SOC Investigation Playbook

Investigation reference for L1, L2, L3 SOC analysts

229 attacks across 30 categories

Each attack has: Severity • What it looks like • L1/L2/L3 checks • TP/FP/BP signals • Containment • Forensics

Severity legend

Critical — immediate IR	High — escalate to L2/L3	Medium — investigate	Low — monitor / triage

Categories

1	**Phishing & Email**	12 attacks	#1–12
2	**Identity & Credentials**	15 attacks	#13–27
3	**Malware & Endpoint**	15 attacks	#28–42
4	**Network & Lateral Movement**	12 attacks	#43–54
5	**Web & Application**	15 attacks	#55–69
6	**Cloud & SaaS**	12 attacks	#70–81
7	**Active Directory & Kerberos**	10 attacks	#82–91
8	**Insider & Data Exfiltration**	8 attacks	#92–99
9	**OT / ICS / IoT**	8 attacks	#100–107
10	**Supply Chain**	6 attacks	#108–113
11	**Mobile**	6 attacks	#114–119
12	**DDoS & Availability**	6 attacks	#120–125
13	**Wireless & Physical**	6 attacks	#126–131
14	**Cryptography & PKI**	6 attacks	#132–137
15	**Container & DevOps**	6 attacks	#138–143
16	**AI & Emerging**	7 attacks	#144–150
17	**Email Security (Advanced)**	6 attacks	#151–156
18	**Database Attacks**	6 attacks	#157–162
19	**API Attacks**	6 attacks	#163–168
20	**Reconnaissance & OSINT**	5 attacks	#169–173
21	**Living-off-the-Land**	6 attacks	#174–179
22	**Defense Evasion**	6 attacks	#180–185
23	**Persistence**	6 attacks	#186–191
24	**Privilege Escalation**	6 attacks	#192–197
25	**Browser Attacks**	6 attacks	#198–203
26	**Social Engineering**	5 attacks	#204–208
27	**Ransomware TTPs**	6 attacks	#209–214
28	**Reverse Shells & C2**	6 attacks	#215–220

29	**Cryptojacking**	4 attacks	#221–224
30	**Deception & Impersonation**	5 attacks	#225–229

Phishing & Email (12)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
1	**Credential Phishing**	**High**	Fake email with link to fake login page that steals password.	• Sender domain real or spoofed? • Link reputation — known bad? • How many users received it? • Attachment hash check	• Anyone clicked the link? • Login from new location after click? • New inbox forwarding rule? • New OAuth app consent?	• Find all recipients across mail logs • Block sender, URL, IP at gateway/proxy/FW • Reset password + revoke sessions for clickers • Extract IOCs, share with TI	TP = bad link + clicked + anomaly. BP = real phish blocked. FP = legit marketing or phish test.	Delete email from all mailboxes. Block sender + URL + IP. Force password reset + new MFA for clickers.	Email + full headers. Browser history of clickers. Mailbox audit logs. Sign-in logs 48h. Endpoint timeline if opened.
2	**Spear Phishing**	**High**	Targeted email crafted with personal context (name, role, project).	• Is the target a high-value role (exec, finance, IT)? • Sender new to user's mail history? • Email body references real internal info? • Link or attachment present?	• Did target interact? • Sender domain registered recently? • Any other targeted users in same dept? • Lure tied to current event/project?	• Treat as targeted attack — assume more attempts coming • Brief target and team on what to watch for • Hunt for similar patterns across org • Coordinate with HR/Legal if exec targeted	TP = personalised lure + bad indicators. BP = real but blocked. FP = legitimate business correspondence.	Same as phishing + add target's mailbox to enhanced monitoring + brief target directly.	Email + headers. OSINT trace on attacker reuse. Target's recent activity. Any earlier pretext emails.
3	**Whaling / Executive Phishing**	**Critical**	Phishing aimed at C-suite or senior executives, often impersonating other execs.	• Target is C-level or board? • Sender impersonates another exec? • Urgency or confidentiality framing? • Request involves money or sensitive data?	• Did exec respond? • Was finance or HR copied? • Any wire transfer or data request triggered? • Lookalike domain (e.g. corp.com vs c0rp.com)?	• Notify CISO and exec immediately • Coordinate with finance to halt any pending action • Forensic preservation of exec mailbox • Legal/PR readiness if breach occurred	TP = exec impersonation + action requested. BP = blocked, no exec interaction. FP = legitimate exec correspondence.	Block lookalike domain everywhere. Brief exec admin/EA. Add exec mailbox to high-priority monitoring. Halt any triggered transactions.	Full email + headers. Exec's recent calendar/travel. Any prior impersonation attempts. Wire/payment system logs.
4	**Business Email Compromise (BEC)**	**Critical**	Attacker uses or impersonates legitimate business email to commit fraud (wire transfer, invoice change).	• Email about money, invoice, or payment changes? • Sender domain matches but slightly off? • Reply-to differs from From address? • Sudden urgency to wire funds?	• Was the legitimate user account compromised? • Any successful login from unusual location? • Inbox rules hiding attacker replies? • Vendor account possibly compromised on their side?	• Coordinate with finance to recover funds (24–72h critical) • File with bank, FBI IC3, or local cybercrime • Reset all credentials, revoke all sessions • Audit all financial communications for last 90 days	TP = funds moved or about to. BP = caught before transfer. FP = legitimate vendor change with proper verification.	Halt all pending transactions. Reset compromised account. Block attacker domain. Notify all finance staff.	Full email thread + headers. Mailbox rules and forwards. Sign-in logs (90 days). Bank transaction logs. Vendor confirmation calls.
5	**Vishing (Voice Phishing)**	**High**	Phone call impersonating IT, bank, or executive to	• User reported a suspicious call? • Caller asked for password, MFA	• Did user share credentials or approve MFA?	• If credentials shared: full account reset	TP = social engineering succeeded or attempted with	Reset credentials. Isolate any endpoint that received remote	Call recording (if available). User's account activity post-

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
			extract info or trigger actions.	code, or remote access? • Caller pressure tactics or urgency? • Number spoofed to look internal?	• Any successful login post-call? • Did user grant remote access (TeamViewer, AnyDesk)? • Voice deepfake suspected (exec impersonation)?	• If remote access granted: isolate endpoint • Brief help desk on the caller pattern • Update vishing training with this example	bad intent. FP = legitimate IT/vendor call. BP = user reported correctly without falling.	access. Block caller number internally. Brief team.	call. Endpoint forensics if remote access used. Phone system logs.
6	**Smishing (SMS Phishing)**	**Medium**	SMS with malicious link, often impersonating delivery, bank, or MFA.	• Sender — known shortcode or random number? • Link domain reputation? • Multiple users got same SMS? • User clicked link?	• Mobile device compromised post-click? • Credentials entered on phishing page? • Banking app or 2FA app interaction? • SIM-swap indicators?	• For corporate phones: MDM scan and clean • User credential reset if entered • Block the URL at corporate proxy • Carrier check for SIM-swap if suspected	TP = malicious SMS + interaction. BP = SMS bad but ignored. FP = legitimate carrier or service SMS.	Block URL. Reset credentials if entered. MDM remediation on device. Educate user.	Screenshot of SMS. Mobile device logs. Phone carrier records. Credential reuse check.
7	**Quishing (QR Code Phishing)**	**Medium**	QR code in email/poster/document leads to phishing page or malware download.	• Email or doc contains a QR code? • Code scans to suspicious URL? • No text URL alongside (evades URL filters)? • Multiple users seen scanning?	• User scanned and visited? • Credentials submitted? • Mobile or desktop scanned? • Same QR seen elsewhere in org?	• Decode and analyse QR destination • Block destination URL/IP • Brief users on QR risks • Update gateway to extract QR URLs from images	TP = malicious QR + scan + interaction. BP = scanned, suspicious, no creds entered. FP = legitimate vendor QR.	Block destination URL. Reset creds if entered. Educate user.	Decoded QR URL. User's scan device logs. Sign-in logs post-scan. Email + image with QR.
8	**OAuth Consent Phishing**	**High**	Email tricks user into granting attacker app permissions to mailbox/data.	• User reported odd consent prompt? • Email pushed user to authorise an app? • App publisher unknown or unverified? • Permissions requested are excessive (mail.read, files.read.all)?	• Was consent granted? Check audit log for app additions • Any data access by the app post-grant? • App appears in other users' tenants? • App requesting offline access (refresh tokens)?	• Revoke app consent across tenant • Hunt for data exfil during access window • Block app's redirect URI • Add app to admin-block list	TP = malicious app granted access. BP = prompt seen, not granted. FP = legit business app onboarding.	Revoke OAuth grant. Block app ID and redirect URI. Reset affected user creds. Audit data access during grant period.	OAuth consent audit logs. App's API call history. Data accessed by the app. User's sign-in around grant time.
9	**Email Spoofing**	**Medium**	Email crafted to appear from someone it isn't (often fails SPF/DKIM/DMARC).	• SPF, DKIM, DMARC results — fail or pass? • Header From vs envelope sender mismatch? • Reply-to domain different from From? • Internal address spoofed externally?	• Volume of similar spoofed mail? • Targeted recipients (finance, HR)? • Linked to a phishing or BEC campaign? • Mail flow rules misconfigured	• Tighten DMARC policy (p=reject) • Add explicit anti-spoof rules for spoofed domain • Notify spoofed party if external • Hunt for downstream impact	TP = spoof + bad payload. BP = spoof caught by gateway. FP = legitimate mail with misconfigured SPF.	Quarantine all mail matching spoof. Update SPF/DKIM/DMARC. Add gateway rules.	Full email headers. Mail gateway logs. DNS records of spoofed domain. Any user interaction logs.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
					allowing it through?				
10	**Email Bombing**	**Medium**	Mass email flood targeting one user, often to hide a real malicious email or DoS the inbox.	• User receiving hundreds/thousands of subscription confirmations? • User is a finance/exec target? • Spike in inbound mail volume to one user? • Any single legitimate-looking email hidden in the flood?	• Was a real attack email sent during the flood? • Any account changes or transactions during flood? • Mailbox quota exceeded — legit mail bouncing? • Source IPs varied (newsletter sites) or single source?	• Identify the cover email — what was the attacker hiding? • Treat user as targeted, escalate monitoring • Mass-unsubscribe or filter the flood • Audit account for unauthorised changes	TP = bombing + cover attack found. BP = bombing only, no follow-up. FP = legitimate newsletter overload.	Filter flood at gateway. Investigate hidden cover attack. Add user to enhanced monitoring.	Mail logs during flood window. Account change logs. Any transactions during flood. Cover email if found.
11	**Mailbox Takeover (Account Compromise)**	**Critical**	Attacker logs into a user's mailbox and operates from inside.	• Sign-in from unusual location/IP/ASN? • New inbox rule (forward, delete, move to RSS)? • Mass send of emails to contacts? • User reports they didn't send recent emails?	• MFA bypassed (token theft) or never configured? • Mail rules hiding replies from user? • OAuth app added during session? • Connected app or eDiscovery activity?	• Force password reset and revoke all sessions • Re-enrol MFA • Remove malicious inbox rules • Hunt for downstream BEC, data exfil, lateral attempts	TP = unauthorised access + actions taken. BP = login blocked at MFA. FP = user travelled, forgot to mention.	Reset password. Revoke all sessions and tokens. Remove rules. Re-enrol MFA. Block source IPs.	Sign-in logs (90 days). Mailbox audit (rules, sends, reads, eDiscovery). All emails sent during compromise. Connected apps.
12	**Reply-Chain Hijack**	**High**	Attacker gets into a real email thread and sends malicious reply that looks legitimate.	• Email is reply to genuine prior thread? • Sender slightly off (lookalike domain)? • Unexpected attachment or link in reply? • Tone or request shift in the latest reply?	• Original participant's mailbox compromised? • Other users in the thread also targeted? • Any clicks/opens by recipients? • Vendor or partner mailbox compromised on their side?	• Notify all thread participants • If internal compromise: full mailbox takeover response • If vendor compromise: notify vendor security • Hunt for similar hijacked threads	TP = hijacked thread with malicious payload. BP = caught before delivery. FP = legitimate reply with attachment.	Block sender + payload. Notify thread members. Vendor security contact if external.	Original thread + hijacked reply. Sender's mailbox logs (if reachable). Recipient interactions. Vendor coordination notes.

Identity & Credentials (15)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
13	**Password Spraying**	**High**	One common password tried across many accounts to avoid lockout.	• Many accounts seeing failed logins from same IP/ASN? • Same password attempt pattern across accounts? • Spread over time (slow spray) or burst? • Any successful login amid the failures?	• Which accounts succeeded — privileged? • Source IP geo + reputation? • MFA enforced on success accounts? • Pattern matches known TTPs (e.g. APT)?	• Reset all targeted accounts that succeeded • Block source IPs/ASN • Force MFA on all accounts • Hunt for post-auth lateral movement	TP = many failures + success. BP = many failures, no success, IP blocked. FP = vuln scanner or pen test.	Block source IPs. Reset succeeded accounts. Force MFA enrolment. Lockout policy review.	Sign-in logs (timestamps, IPs, accounts, results). Source IP attribution. Post-success actions for breached accounts.
14	**Credential Stuffing**	**High**	Bulk-tested credentials from a third-party breach against your login.	• High volume of distinct username+password attempts from one IP? • Mix of success/failure (3rd-party breach pattern)? • Source IP from anonymising service or botnet? • User-agent unusual or rotating?	• Which accounts succeeded? • Are succeeded passwords found on HIBP / breach feeds? • MFA stopped some attempts? • Multiple source IPs (botnet) or single?	• Reset all succeeded accounts • Block source IPs/ASN • Roll out password breach checking • Push MFA enforcement	TP = bulk distinct creds + succeeded logins. BP = volume seen, all blocked. FP = legit user with many failed attempts.	Block source IPs. Reset compromised accounts. Enforce MFA. Implement breach-password blocking.	Sign-in logs. IP attribution. Comparison with breach databases. Post-login activity on succeeded accounts.
15	**Brute Force**	**Medium**	Many password guesses against one account.	• High failure count on single account? • Source IP — internal or external? • Locked out yet? • Account is privileged or service account?	• Did any attempt succeed? • Source same as recent recon? • Account a target of past attacks? • MFA in place?	• Reset and harden if succeeded • Block source IP • Add account to enhanced monitoring • Review lockout policies	TP = brute force + success. BP = brute force, account locked, no success. FP = user genuinely forgetting password.	Block source. Reset password. Force MFA. Tighten lockout policy.	Sign-in logs for the account. IP attribution. Account history. Any post-success activity.
16	**MFA Fatigue / Push Bombing**	**High**	Attacker has password, sends many MFA push prompts hoping user approves.	• Multiple MFA prompts to one user in short window? • User reported unexpected MFA notifications? • Source of auth attempts — unusual location? • Did user approve any prompt?	• When user approved, what location was the auth from? • Account activity post-approval? • Other accounts seeing same pattern? • Number-matching MFA in use or just push?	• Revoke session, reset password, re-enrol MFA • Move to number-matching or FIDO2 • Brief user on MFA fatigue tactic • Hunt for lateral movement	TP = many prompts + user approved + foreign auth. BP = prompts seen, user denied all. FP = user actually logging in many times.	Revoke sessions. Reset password. Re-enrol MFA with number-matching. Block source IP.	MFA logs (prompts, approvals, denials). Sign-in logs around approval. Endpoint logs if compromise followed.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
17	**Adversary-in-the-Middle (AitM) Phishing**	**Critical**	Phishing page proxies real login, captures session token after MFA — bypasses MFA.	• Phishing email or link reported? • Login page is proxy (Evilginx, EvilProxy patterns)? • User entered creds AND completed MFA on fake page? • Sign-in immediately after from different location?	• Session token replayed from attacker IP? • Same session active in two geos? • Mailbox actions post-token-theft? • OAuth grants or rule additions?	• Revoke all sessions and refresh tokens for user • Reset password and re-enrol MFA • Hunt for downstream actions (BEC, data exfil) • Add detection for impossible-session-replay	TP = MFA passed but session used elsewhere. BP = phish blocked before submission. FP = user with multiple legitimate devices.	Revoke all tokens (not just sessions). Password reset. New MFA. Block attacker infrastructure. Conditional access tightening.	Sign-in logs (token issue + replay events). Mailbox audit. OAuth grants. Endpoint/browser logs. Phishing infrastructure analysis.
18	**Session Hijacking / Cookie Theft**	**High**	Attacker steals browser session cookie/token and uses it without re-authenticating.	• Sign-in event without password/MFA event? • Same session ID seen from two IPs? • Infostealer alert on user's endpoint? • User reported lost laptop or device?	• Browser logs show cookie export or extension abuse? • Endpoint had infostealer (RedLine, Lumma, Vidar)? • Token replay timeline? • Post-replay actions in the application?	• Force token revocation • Wipe and reissue user device if endpoint compromised • Hunt for what attacker did with session • Roll out token-binding / phishing-resistant MFA	TP = token used from foreign IP, no auth event. BP = session expired before exploit. FP = legitimate roaming.	Revoke all tokens. Reimage endpoint. Reset all credentials user had cached. Audit data accessed during session.	Token logs. Endpoint forensics (browser cookies, malware). Sign-in logs. Application action logs during replay.
19	**Pass-the-Hash**	**High**	Attacker uses stolen NTLM hash to authenticate without knowing the password.	• NTLM auth events from unusual source? • Account used in lateral movement pattern? • Logon type 3 (network) from workstation rather than server? • Privileged account auth where it shouldn't be?	• LSASS access events on source endpoint? • Mimikatz or similar tool seen on host? • Hash reused across multiple targets? • Domain controller auth volume spike?	• Reset compromised account password (twice for KRBTGT-class) • Hunt for further lateral movement • Force tier-0 account isolation • Disable NTLM where possible	TP = hash reuse confirmed by patterns. BP = LSASS access blocked. FP = legitimate admin tool using NTLM.	Reset password twice. Isolate source host. Restrict NTLM. Tier the privileged accounts.	4624 (logon type 3) events. LSASS access logs. Endpoint EDR for credential dumping tools. Lateral movement timeline.
20	**Pass-the-Ticket**	**High**	Attacker uses stolen Kerberos ticket (TGT or service ticket) to authenticate.	• Unusual Kerberos auth (4768/4769) for an account? • Ticket used from host that didn't request it? • Logon without preceding password validation?	• Mimikatz or Rubeus signatures on source host? • Golden Ticket indicators (long lifetime, anomalous PAC)?	• Reset KRBTGT account twice (full domain remediation) • Reset all admin and service accounts	TP = ticket anomalies + lateral use. BP = ticket extraction tool blocked. FP = trusted forwarded auth (cross-realm).	KRBTGT double-reset. Reset all privileged accounts. Image source host. Tier-0 lockdown.	Kerberos events 4768/4769/4770. Endpoint EDR for Mimikatz/Rubeus. DC logs. PAC validation logs.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• Account is high-privilege?	• Silver Ticket (service-specific)? • Domain controller compromise suspected?	• Forensic image of source host • Full AD security review			
21	**Kerberoasting**	**High**	Attacker requests service tickets for SPN-enabled accounts and cracks them offline.	• Spike in 4769 events (service ticket requests) from one account? • Many SPN service tickets requested in short time? • Encryption type RC4 (weak, crackable)? • Source account is non-admin user?	• Which SPN accounts targeted? • Did attacker successfully use any cracked password? • Service accounts with weak passwords identified? • Tools like Rubeus or GetUserSPNs.py traces?	• Reset cracked service account passwords (long, complex) • Move service accounts to Group Managed Service Accounts (gMSA) • Disable RC4, force AES • Hunt for tools and persistence	TP = bulk SPN ticket requests + cracking activity. BP = requests caught early. FP = legitimate SPN enumeration by admin tool.	Reset service account passwords. Move to gMSA. Disable RC4. Block source.	4769 events with details. Endpoint EDR for Kerberoasting tools. Service account password complexity audit.
22	**AS-REP Roasting**	**Medium**	Attacker requests AS-REP for accounts with Kerberos pre-auth disabled, cracks offline.	• AS-REQ events for accounts with no pre-auth? • Multiple accounts queried from one source? • Account is service or legacy? • Source not normal workstation?	• Which accounts have pre-auth disabled (and why)? • Did attacker crack and reuse the credential? • Tool indicators (Rubeus, Impacket)? • Persistence following success?	• Re-enable pre-auth on all flagged accounts • Reset passwords on those accounts • Hunt for lateral movement • Document why any account legitimately needs no pre-auth (rare)	TP = AS-REP enumeration + cracked use. BP = enum without success. FP = legitimate legacy app needing it.	Re-enable pre-auth. Reset passwords. Block source. Audit accounts.	4768 events. Pre-auth flag audit. Endpoint EDR. Cracked credential reuse logs.
23	**Golden Ticket Attack**	**Critical**	Attacker forges a Kerberos TGT using stolen KRBTGT hash — full domain compromise.	• Anomalous TGT lifetime (10 years) or no AS-REQ before TGS? • High-privilege account not in domain admin? • Logon without prior password validation? • Strange account name in PAC?	• KRBTGT compromise indicators? • DC compromise suspected (DCSync alerts)? • Forged ticket used for sensitive resource access? • Cross-tier movement	• KRBTGT password reset TWICE (24h apart) • Full AD compromise response — likely rebuild • Reset all admin accounts	TP = ticket forgery confirmed. BP = KRBTGT access blocked. FP = none (this is always serious).	KRBTGT double-reset. Domain rebuild planning. Tier-0 isolation. All admin reset.	All DC logs. KRBTGT password change history. Kerberos events. EDR on suspected attacker hosts. PAC analysis.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
					(workstation to DC)?	• Forensic preserve DCs			
24	**Silver Ticket Attack**	**High**	Attacker forges service ticket using stolen service account hash — access to specific service.	• Service ticket usage without preceding TGS request? • Anomalous PAC for the service? • Service account password changed recently (not by IT)? • Privileged service access by non-privileged user?	• Which service compromised (SQL, CIFS, HTTP)? • Service account hash extraction trace? • Lateral movement from compromised service? • Privilege escalation via service?	• Reset service account password (long, complex) • Move to gMSA • Hunt for lateral movement from service • Audit service permissions	TP = forged ticket pattern + service access. BP = hash extraction blocked. FP = legitimate service auth pattern.	Reset service account. gMSA migration. Block source. Audit service access.	Service-specific access logs. Kerberos events. Endpoint EDR for hash extraction. Service ACL audit.
25	**DCSync Attack**	**Critical**	Attacker mimics a domain controller and requests password hashes via replication.	• Replication request from non-DC source? • Account performing DCSync is not domain admin? • Event 4662 with DS-Replication-Get-Changes-All? • Mimikatz or DCSync tool indicators?	• Which accounts had hashes pulled (likely all)? • Source endpoint compromise scope? • Earlier privilege escalation chain? • Persistence post-DCSync?	• Treat as full AD compromise — KRBTGT reset twice • Reset all admin and service accounts • Rebuild trust posture • Forensic image source endpoint	TP = DCSync from non-DC. BP = blocked at firewall/permissions. FP = legitimate AD migration tool.	Full AD compromise response. KRBTGT double-reset. All privileged reset. Endpoint isolation.	4662 events with replication-related GUIDs. DC logs. Source endpoint full forensics. AD permissions audit.
26	**LDAP Reconnaissance**	**Low**	Attacker queries Active Directory to map users, groups, computers.	• High volume of LDAP queries from one source? • Source not a typical enumeration tool host? • Queries for sensitive groups (Domain Admins)? • User account doing the queries?	• Tools like BloodHound, SharpHound, ADExplorer trace? • Followed by lateral movement? • Source endpoint compromised? • Earlier credential theft?	• If source compromised: full IR • Block source if external • Tighten LDAP query monitoring • Hunt for what they were targeting next	TP = enum + post-recon attack. BP = enum without follow-up (still suspicious). FP = legitimate admin tool.	Block source. Investigate endpoint. Reset credentials if compromised.	LDAP query logs. Source endpoint EDR. Network traffic to DCs. Account activity timeline.
27	**Credential Dumping (LSASS)**	**Critical**	Attacker reads LSASS process memory to extract passwords/hashes/tickets.	• LSASS access alert from EDR? • Tool: Mimikatz, ProcDump, Task Manager dump? • User context — admin or SYSTEM? • Source process unusual (not procexp, not legit tool)?	• Was dump successful? • Subsequent lateral movement? • File written to disk (.dmp) or in-memory? • Multiple hosts targeted?	• Reset all credentials cached on host • Isolate and image host • Hunt for lateral movement • Enable Credential	TP = LSASS read by non-system process. BP = EDR blocked the access. FP = legitimate admin tool (rare, justify).	Isolate host. Reset all creds cached. Block source. Roll out Credential Guard.	EDR LSASS access logs. Process tree. Files written. Lateral movement evidence. Memory snapshot if possible.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
						Guard / LSA protection			

Malware & Endpoint (15)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
28	**Ransomware**	**Critical**	Files encrypted with ransom note left behind.	• EDR alert for mass file encryption? • Ransom note files (README, HOW_TO_DECRYPT)? • Shadow copies deleted (vssadmin)? • Multiple hosts affected?	• Initial access vector (phish, RDP, exploit)? • Lateral movement before encryption? • Data exfiltration before encryption (double extortion)? • Backup systems also targeted?	• Activate IR plan, engage IR retainer • Isolate all affected segments • Coordinate with legal, leadership, possibly LE • Recover from clean backups, do not pay if avoidable	TP = encryption + ransom note. BP = ransomware blocked at execution. FP = legit file encryption tool (rare).	Network isolate all affected hosts. Disable AD accounts of affected users. Block C2 IPs/domains. Preserve evidence.	EDR timeline. Ransom note + encrypted file samples. Initial access logs. Lateral movement trace. Exfil DNS/network logs.
29	**Infostealer**	**High**	Malware that harvests passwords, cookies, crypto wallets, and exfiltrates them.	• EDR detection for known stealer (RedLine, Lumma, Vidar, Raccoon)? • Browser data accessed by suspicious process? • Outbound connection to known C2? • User reported odd download or game crack?	• What credentials were taken (browser, app, vault)? • Session cookies stolen — token replay risk? • Which apps had saved creds? • Lateral spread or single-host?	• Reset all credentials user had on the host • Revoke all sessions for user • Reimage endpoint • Watch for credential reuse externally (HIBP)	TP = stealer detected + exfil event. BP = stealer blocked pre-exec. FP = security tool's own credential test.	Isolate host. Reimage. Reset all user creds. Revoke tokens. Block C2.	EDR full timeline. Process tree. Files written. Network connections. Browser profile state. Stolen-creds-on-darkweb monitoring.
30	**Remote Access Trojan (RAT)**	**Critical**	Persistent backdoor giving attacker hands-on-keyboard control.	• EDR detection for known RAT (Cobalt Strike, Sliver, NetWire, Quasar)? • Outbound beaconing pattern (regular intervals)? • Process injection or living-off-the-land binary? • User unaware of remote sessions?	• C2 infrastructure analysis • Persistence mechanism (registry, service, task)? • Lateral movement from infected host? • Credential access on the host?	• Full IR — assume hands-on attacker present • Isolate immediately, do not just block C2 • Hunt across estate for same TTPs • Engage threat intel for attribution	TP = RAT process + C2 + interactive activity. BP = RAT delivery blocked. FP = legitimate remote admin tool.	Isolate host immediately. Reimage. Reset all creds. Block C2 globally. Hunt for sister implants.	EDR timeline. C2 traffic capture. Process tree. Persistence artefacts. Memory image. Lateral movement logs.
31	**Banking Trojan / Loader**	**High**	Malware (Emotet, TrickBot, IcedID) that steals banking creds and drops further malware.	• EDR detection for loader family? • Phishing email with macro doc as initial vector? • Outbound to known loader C2? • Subsequent payload dropped (often ransomware)?	• What was the second-stage payload? • Lateral movement (SMB, WMI)? • Banking app or finance system targeted? • Persistence and scheduled tasks?	• Treat as ransomware precursor — accelerate response • Isolate, reimage, reset credentials • Hunt for second-stage across estate • Block all related C2	TP = loader + payload activity. BP = loader blocked at delivery. FP = legitimate downloader (rare).	Isolate. Reimage. Reset creds. Block C2. Look for second-stage on other hosts.	EDR timeline. Email + macro analysis. Payload samples. C2 captures. Persistence artefacts.
32	**Cryptominer**	**Medium**	Malware uses victim CPU/GPU to mine cryptocurrency.	• Sustained high CPU on host? • Outbound connections to known mining pools (stratum protocol)?	• How did it get there (web exploit, supply chain, insider)? • Persistence configured?	• Reimage host • Block mining pool IPs/domains • Investigate root cause and patch	TP = miner + outbound to pool. BP = miner blocked at execution. FP = legitimate	Kill miner process. Reimage if persistent. Block mining	EDR timeline. Network connections to pools. Wallet addresses. Persistence

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• Process names like xmrig, cgminer, NiceHash? • Performance complaints from user?	• Spread to other hosts? • Cloud workload also affected?	• Audit any cloud bills for hijacked compute	compute workload (rare).	pools. Patch root cause.	mechanism. Initial access vector.
33	**Wiper Malware**	**Critical**	Destructive malware that erases data with no recovery option.	• EDR detection for known wiper (NotPetya, Olympic Destroyer, HermeticWiper)? • Mass file deletion or MBR overwrite? • Multiple hosts affected simultaneously? • Geopolitical context (nation-state activity)?	• Initial access vector? • Self-propagation mechanism? • Targeted vs collateral damage? • Other regions/business units hit?	• Activate full IR + crisis management • Engage law enforcement • Restore from offline backups • Public communications coordination	TP = data destruction confirmed. BP = wiper blocked. FP = none — always treat as serious.	Isolate everything affected. Stop propagation (block SMB, kill spreading process). Engage IR retainer.	Affected host images. Wiper sample. Network propagation logs. Initial access trace. Geopolitical attribution context.
34	**Rootkit / Bootkit**	**Critical**	Malware that hides below the OS level (UEFI, kernel) and survives reboots.	• EDR alert for kernel driver loading? • Boot integrity check failure (Secure Boot)? • Anomalous kernel modules or drivers? • System exhibiting weird behaviour AV cannot explain?	• UEFI / firmware tampering signs? • Hidden processes (compare ps vs ETW)? • Network traffic without process attribution? • Vendor-specific tools detecting it (LoJax, MosaicRegressor)?	• Cannot trust the host — full firmware reflash + OS reimage • Investigate supply chain or physical access vector • Hunt for similar across fleet • Engage advanced forensics	TP = below-OS persistence confirmed. BP = bootkit blocked at install. FP = legitimate vendor driver (rare).	Disconnect host. Plan firmware reflash + reimage (or replace hardware). Treat as advanced threat.	Memory image. Disk forensics. UEFI dump. Network captures. Often requires vendor or specialist help.
35	**Worm / Self-Propagating Malware**	**High**	Malware that spreads itself across network without user interaction.	• Multiple hosts showing same alert in short time? • SMB/RPC traffic spike between internal hosts? • Same process name appearing on many hosts? • Vulnerability being exploited (e.g. SMBv1, EternalBlue)?	• Propagation mechanism (SMB, RDP, WMI, exploit)? • Patch level on affected vs unaffected hosts? • Worm payload — what does it do besides spread? • Network segmentation effective?	• Network-level containment (block SMB at segment boundaries) • Patch the exploited vulnerability across estate • Reimage all affected hosts • Validate segmentation	TP = same malware on multiple hosts spreading. BP = first host caught, no spread. FP = mass legitimate software deployment.	Segment-level network isolation. Disable propagation protocol. Patch. Reimage.	Patient zero identification. Network spread map. Sample analysis. Patch-level audit.
36	**Fileless Malware (In-Memory)**	**High**	Malicious code runs entirely in memory, no executable on disk — hard to detect.	• Suspicious PowerShell/WMI/Office macro execution? • Anomalous process behaviour without disk artefact? • Memory-resident process spawned by Office? • No file but EDR sees malicious in-memory activity?	• Parent process chain (e.g. winword.exe → powershell.exe)? • Encoded or obfuscated commands? • Persistence via WMI subscription or scheduled task? • Lateral movement via WMI/PSRemoting?	• Hunt across estate for same TTPs • Memory image for analysis • Reimage affected hosts • Tighten script-block logging and AMSI	TP = malicious in-memory + parent chain. BP = blocked by AMSI/EDR. FP = legitimate admin script.	Isolate. Memory dump. Reimage. Tighten logging. Block C2.	Memory image. ETW + Sysmon logs. PowerShell script-block logs. Parent process chain. WMI subscriptions.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
37	**Macro Malware (Office Documents)**	**Medium**	Malicious VBA macro in Word/Excel document executes payload when opened.	• Office document opened and triggered process spawn? • Macro auto-enabled or user clicked Enable? • Outbound connection from Office process? • Document came from external email?	• Macro contents (deobfuscated)? • Payload downloaded — what is it? • User clicked through warnings? • Other recipients of same doc?	• Block macro source domain • Disable macros from internet (group policy) • Hunt for second-stage payload • Brief users	TP = macro + payload + C2. BP = macro blocked. FP = legitimate business macro.	Block source. Disable internet macros via GPO. Hunt for payload across hosts.	Document + macro extraction. EDR timeline. C2 traffic. Payload analysis.
38	**Living-off-the-Land (LOLBin) Abuse**	**High**	Attacker uses legitimate Windows binaries (PowerShell, certutil, mshta, rundll32) to evade detection.	• LOLBin spawned with unusual arguments (download, decode, execute)? • Parent process unusual (e.g. winword spawning powershell)? • Network connection from LOLBin? • Encoded base64 or hidden cmdline?	• What did LOLBin actually do (decode, download, execute)? • Multiple LOLBins chained? • Persistence created via LOLBin? • Lateral movement using LOLBin?	• Hunt for LOLBin abuse patterns across estate • Tighten EDR rules for LOLBin command-line patterns • Application whitelisting on critical hosts • Block specific bad uses (e.g. certutil -urlcache)	TP = LOLBin + bad cmdline + bad parent. BP = blocked by behavioural detection. FP = admin script using legit utility.	Isolate. Block C2. Tighten EDR. Reset creds.	EDR command-line logs. Sysmon Event 1. Parent-child process tree. Network connections. PowerShell logs.
39	**USB / Removable Media Malware**	**Medium**	Malware delivered via USB drive — autorun or social engineering.	• USB device insertion event? • Autorun or shortcut execution from USB? • Process spawned from removable drive letter? • User found a USB and plugged it in (BadUSB)?	• What was on the USB (sample)? • User's role (high-value target)? • Other USBs from same source? • Air-gapped systems compromised?	• Reimage host • Disable USB autorun via GPO • Block mass-storage USB on critical systems • Brief user	TP = USB + malware execution. BP = USB blocked by policy. FP = legitimate vendor USB.	Isolate host. Reimage. Disable USB autorun. Restrict USB usage policy.	USB device logs (vendor, serial, time). EDR timeline. File samples from USB. User interview.
40	**Trojanised Software / Cracked Apps**	**High**	Pirated software, fake installers, or game cracks bundled with malware.	• User downloaded software from non-corporate source? • Installer signed by unknown publisher? • EDR detected malware during install? • Outbound C2 post-install?	• What software (game, productivity, crack)? • Malware family? • Persistence and additional payloads? • User's role and access?	• Reimage • User awareness coaching + policy reminder • Tighten download restrictions • Watch for credential reuse	TP = trojan + execution. BP = blocked at download/install. FP = legitimate vendor with reputational lag.	Reimage. Reset creds. Block source domain. Restrict admin install rights.	Installer sample. EDR timeline. Browser download logs. C2 traffic.
41	**Process Injection**	**High**	Malicious code injected into legitimate process to evade detection.	• EDR alert for injection technique (DLL injection, hollowing, APC)? • Legitimate process (svchost, explorer) doing unusual things? • Network connection from unexpected process?	• Source of injection (parent process)? • Payload identified? • Persistence? • Lateral movement?	• Isolate, reimage • Hunt for same injection patterns across estate • Tune EDR for the technique • Reset creds	TP = injection + payload. BP = injection blocked. FP = legitimate AV/EDR action.	Isolate. Reimage. Block C2. Reset creds.	EDR injection event details. Memory image. Parent-child chain. Payload sample.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• Suspended process being written to?					
42	**DLL Hijacking / Sideloading**	**Medium**	Malicious DLL placed where legitimate app loads it from, executing attacker code.	• Legitimate app loading DLL from unexpected path? • DLL signed differently than expected? • DLL path writable by user? • Process spawning behaviour after DLL load?	• Which application abused? • Persistence? • Same DLL on other hosts (mass deployment)? • Patch / vendor advisory exists?	• Remove malicious DLL • Patch app or update load order • Hunt across fleet • Tighten file integrity monitoring	TP = malicious DLL loaded by legit app. BP = blocked. FP = legitimate plugin DLL.	Remove DLL. Reimage if persistence. Patch.	DLL sample + signature analysis. Application config. EDR load events. File system path permissions.

Network & Lateral Movement (12)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
43	**Lateral Movement via SMB**	**High**	Attacker uses SMB shares (admin$, c$) to move between hosts.	• SMB connections from workstation to workstation (unusual)? • Admin$ share access from non-admin host? • File copy to remote ADMIN$? • Followed by remote service or task creation?	• Source compromised endpoint? • Tools (PsExec, SMBexec)? • Credentials being reused (pass-the-hash signs)? • Spread pattern across many hosts?	• Block SMB lateral at segment boundaries • Reset reused credentials • Hunt for source compromise • Isolate spread hosts	TP = unusual SMB lateral + remote exec. BP = blocked by segmentation. FP = legitimate IT remote management.	Network isolate affected segment. Block SMB workstation-to-workstation. Reset creds.	4624 logon type 3 events. Sysmon network events. Source endpoint EDR. Files copied to ADMIN$.
44	**Lateral Movement via RDP**	**High**	Attacker uses RDP to move between hosts interactively.	• RDP from internal host to internal host (unusual)? • Account is not admin/IT but using RDP? • Multiple RDP attempts to many hosts? • Time-of-day unusual?	• Interactive logons (4624 type 10) from unexpected sources? • Tools running on RDP target post-login? • Compromised credentials used? • RDP from internet (exposed RDP)?	• Disable RDP where not needed • Force MFA on RDP (RDG/jump host) • Reset compromised creds • Hunt across estate	TP = RDP lateral + post-RDP attack actions. BP = blocked at jump host. FP = legitimate IT remote work.	Block RDP at segment. Force jump-host model. Reset creds. Isolate hosts.	4624 type 10 events. RDP session logs. Source/dest pairing. Post-login activity.
45	**WMI / WinRM Lateral Movement**	**High**	Attacker uses WMI or WinRM to remotely execute commands.	• WMI/WinRM events from unexpected source? • Process spawned via wmic or Invoke-Command? • No interactive logon, just remote exec? • Encoded command or payload?	• Source compromised? • Persistence via WMI subscription? • Multiple targets? • Credential reuse pattern?	• Block WMI/WinRM at segment boundaries • Hunt for WMI persistence • Reset creds • Investigate source	TP = remote WMI/WinRM exec from anomalous source. BP = blocked at FW. FP = legitimate admin automation.	Block WMI/WinRM lateral. Remove WMI persistence. Reset creds.	WMI activity logs. WinRM logs. Source endpoint EDR. Encoded command analysis.
46	**Beaconing / C2 Communication**	**Critical**	Compromised host calls home to attacker server at regular intervals.	• Outbound connections at fixed intervals (every 60s, every hour)? • Same destination repeated, low data volume per beacon? • User-agent or destination on TI feed? • Multiple hosts hitting same destination?	• Process making the connection? • Beacon decoded — Cobalt Strike / Sliver / Mythic patterns? • Jitter and sleep configured? • Egress via uncommon protocol (DNS, ICMP)?	• Block C2 globally • Isolate beaconing hosts • Hunt for sister implants • Engage threat intel	TP = periodic C2 + bad destination. BP = blocked at proxy. FP = legitimate periodic update check.	Isolate hosts. Block C2 IPs/domains. Reimage. Reset creds.	NetFlow / proxy logs. Beacon sample. EDR process tree. Memory image. C2 IOCs for hunt.
47	**DNS Tunnelling**	**High**	Attacker exfiltrates data or commands	• High volume of DNS queries to one domain?	• Domain age and reputation?	• Block the domain at DNS	TP = DNS tunnel patterns + bad domain. BP =	Block domain. Force internal resolver.	DNS query logs. EDR on source host.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
			through DNS queries.	• Long random subdomains in queries? • TXT, NULL or unusual record types in volume? • Source not a normal DNS-heavy app?	• Tools like dnscat2, iodine traces? • Data volume implied by query count? • Same pattern other hosts?	• Force internal DNS resolver only (no direct port 53) • Investigate compromised host • DNS query monitoring tightened	blocked by DNS firewall. FP = legitimate DNS-based service (rare).	Isolate host. Reimage.	Decoded payload from DNS. Domain WHOIS.
48	**Domain Fronting / CDN Abuse**	**High**	Attacker hides C2 traffic behind legitimate CDN domain.	• TLS SNI and HTTP Host header mismatch? • Outbound to major CDN (CloudFront, Fastly) but unusual app? • Volume or pattern not matching legit CDN use? • TI feed flags hidden domain?	• Decrypt inspection (TLS proxy) showing real backend? • Process initiating? • Subdomain or path patterns indicating C2? • Multiple hosts using same CDN endpoint?	• Block at TLS proxy with deeper inspection • Restrict CDN egress to known apps • Hunt for similar patterns • Isolate hosts	TP = SNI/Host mismatch + C2 patterns. BP = blocked by TLS inspection. FP = legitimate CDN-hosted service.	Block real backend domain. TLS inspection enforcement. Isolate hosts.	TLS proxy logs (decrypt). NetFlow. EDR on source. Domain analysis.
49	**Port Scanning / Network Reconnaissance**	**Low**	Attacker probes hosts and ports to map the network.	• Source IP hitting many ports on one host or many hosts? • Same source seen scanning before? • External or internal source? • Followed by connection attempts to open ports?	• Tool signature (Nmap, Masscan)? • Targeted scan or broad? • Internal source compromised? • Reconnaissance for known exploit?	• Block external scanners • Investigate internal source for compromise • Tighten firewall ACLs • Hunt for exploitation post-recon	TP = scan + exploitation attempt. BP = scan only, blocked. FP = vuln scanner or pen test.	Block source IP. Investigate internal source. Tighten egress.	Firewall logs. NetFlow. Source attribution. Targeted ports/services.
50	**ARP Spoofing / Local MitM**	**Medium**	Attacker on local network impersonates gateway to intercept traffic.	• Multiple MAC addresses for same IP on switch? • Gratuitous ARP replies from unexpected source? • Hosts reporting connection issues? • Switch logs showing ARP storms?	• Source MAC vendor (rogue device)? • Tools (ettercap, bettercap) traces? • Wireless or wired entry? • What traffic intercepted?	• Identify and physically remove rogue device • Implement DHCP snooping and dynamic ARP inspection • Network segmentation review • Investigate insider/visitor	TP = ARP spoof + traffic interception. BP = caught and blocked. FP = legitimate HA failover.	Disable rogue switch port. Implement DAI. Investigate physical access.	Switch ARP/MAC tables. Packet capture. Physical access logs. Camera footage.
51	**DHCP Spoofing**	**Medium**	Rogue DHCP server on network hands out malicious config (gateway, DNS).	• Hosts getting unexpected DHCP config (different gateway/DNS)? • Multiple DHCP offers seen on segment? • Users reporting weird routing?	• Source MAC of rogue DHCP? • Wireless or wired? • Malicious DNS pushed? • Traffic redirection occurred?	• Disable rogue port • DHCP snooping enforcement • Investigate physical access • Reset affected hosts' configs	TP = rogue DHCP + bad config. BP = caught by DHCP snooping. FP = legitimate test DHCP server.	DHCP snooping. Disable rogue port. Reset host configs. Investigate.	Switch DHCP logs. Rogue device MAC. DNS records pushed. Affected hosts.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• Switch logs flagging rogue DHCP?					
52	**DNS Cache Poisoning / Spoofing**	**High**	Attacker corrupts DNS resolver cache to redirect users to malicious sites.	• Users reporting wrong site loads? • DNS resolver logs showing unexpected answers? • TLS cert mismatches? • Multiple users affected on same resolver?	• Source of poisoning (resolver, upstream, MitM)? • DNSSEC enabled? • Specific domains targeted? • Phishing or malware delivery via redirect?	• Flush DNS caches • Patch / upgrade resolver • Enable DNSSEC validation • Investigate downstream compromise	TP = poison + user impact. BP = blocked by DNSSEC. FP = legitimate DNS change misdiagnosed.	Flush caches. Patch resolver. DNSSEC enforce. Investigate downstream.	DNS resolver logs. Affected user reports. TLS cert errors. Network captures.
53	**VPN Compromise**	**High**	Attacker uses stolen VPN credentials to access internal network.	• VPN login from unusual country/ASN? • User's account active from two locations? • No MFA on VPN or MFA bypassed? • Lateral movement post-VPN?	• Credentials sourced from infostealer or breach? • Unusual hours of VPN? • Endpoint posture check failing? • Volume of connections from same source?	• Reset VPN cred + force MFA + posture check • Block source IP • Hunt for what attacker did inside • Review VPN logs 90 days	TP = anomalous VPN login + lateral. BP = blocked at MFA. FP = user travelling.	Reset cred. Revoke session. Force MFA. Block IP. Hunt internal lateral.	VPN auth logs. Source IP attribution. Internal traffic from VPN IP. Endpoint connect activity.
54	**Tunnelling via SSH or Reverse Proxy**	**High**	Attacker uses SSH or reverse-proxy tools (ngrok, frp, Chisel) to bridge into internal network.	• Outbound SSH from unusual host? • Connection to known tunnelling service? • Process running ngrok/frp/Chisel binaries? • Persistent outbound tunnel pattern?	• Tunnel direction (in or out)? • Service exposed via tunnel? • Tunnel destination attribution? • Other hosts running same tools?	• Block tunnelling service domains • Application whitelisting on critical hosts • Investigate source • Hunt for tools	TP = unauthorised tunnel from internal. BP = caught at egress. FP = approved DevOps tunnel.	Block tunnel destinations. Kill tunnel process. Reimage. Reset creds.	Process logs. Network connections. Tunnel binary samples. Egress logs.

Web & Application (15)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
55	**SQL Injection (SQLi)**	**High**	Attacker injects SQL through input fields to read or modify the database.	• WAF alert for SQLi pattern (UNION SELECT, OR 1=1)? • Source IP making many odd requests to DB-backed endpoint? • HTTP 500 errors with SQL traces? • Request payload contains SQL keywords?	• Did any request bypass WAF? • Database error responses revealing schema? • Data exfiltrated (response sizes)? • Authentication bypass via SQLi?	• Patch the vulnerable endpoint (parameterised queries) • Audit DB for unauthorised access • Review data potentially exfiltrated • Force WAF rules tightening	TP = SQLi + successful response or data leak. BP = WAF blocked. FP = legit query containing SQL keywords.	Block source IP. Patch app. Tighten WAF. Database access review.	WAF logs. Web server logs. DB query logs. App error logs. Source IP attribution.
56	**Blind SQL Injection**	**High**	SQLi where attacker infers data from response timing or boolean differences.	• Many similar requests with slight payload variations? • Time-based queries (SLEEP, WAITFOR)? • Boolean inference patterns? • Slow response times correlated with payloads?	• Tools (sqlmap) signatures? • Database fingerprinting patterns? • Successful data inference? • WAF bypass techniques used?	• Patch endpoint • Audit data access • Tighten WAF for time-based and boolean patterns • Source IP block	TP = blind SQLi pattern + inference success. BP = blocked. FP = legitimate slow queries.	Block source. Patch. WAF tightening.	WAF + web logs. Response time analysis. DB query logs. sqlmap signature match.
57	**Cross-Site Scripting (XSS)**	**Medium**	Malicious JS injected into web page, runs in other users' browsers.	• WAF alert for script injection? • User reports unexpected page behaviour? • Reflected, stored, or DOM-based? • Cookie theft or redirect attempted?	• Did script execute in real users' browsers? • Cookies/tokens stolen? • Account compromise downstream? • Source of injection (input field, URL param)?	• Patch app (output encoding, CSP) • Audit affected users • Reset compromised accounts • Tighten WAF	TP = stored XSS + execution. BP = WAF blocked. FP = legitimate HTML/JS in input.	Patch. Remove stored payload. Tighten CSP. Reset affected users.	Web logs. WAF logs. DB content (stored XSS). User session logs. Browser console errors.
58	**Cross-Site Request Forgery (CSRF)**	**Medium**	Trick authenticated user's browser into making unwanted request to a site.	• Unexpected state-changing request from user? • No CSRF token or invalid token? • Referrer/Origin mismatch? • User reports actions they didn't take?	• Was malicious request successful? • User clicked link from external site? • SameSite cookie configured? • Token validation logic bypassed?	• Patch app (proper CSRF tokens, SameSite cookies) • Audit affected accounts • Brief users on suspicious links • Tighten WAF	TP = CSRF triggered action without consent. BP = blocked by token check. FP = legit cross-site flow.	Patch. Roll back affected actions. SameSite enforcement.	Web logs. Referrer/Origin analysis. User session logs. Action audit.
59	**Server-Side Request Forgery (SSRF)**	**High**	Attacker tricks server into making requests to internal or cloud metadata services.	• WAF alert for SSRF pattern (localhost, 169.254.169.254, internal IP)?	• Did request reach internal service or metadata? • Cloud credentials extracted via metadata?	• Patch app (URL allow-list, deny internal/metadata) • Rotate cloud credentials if metadata accessed	TP = SSRF + internal access. BP = blocked. FP = legitimate webhook to allowed endpoint.	Patch. Block metadata endpoint at app tier. Rotate credentials.	Web logs. App logs. Cloud audit (metadata calls). Internal access logs.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• User input reaches outbound HTTP function? • Cloud metadata endpoint access from app? • Internal service access from web tier?	• Internal data exfiltrated? • WAF bypass techniques?	• Tighten egress from app tier • Audit data access		Egress restriction.	
60	**XML External Entity (XXE)**	**High**	Malicious XML with external entity references to read files or reach internal services.	• WAF alert for XML with DOCTYPE or ENTITY? • XML parser in use on the endpoint? • Outbound from app tier triggered by XML upload? • File contents in response?	• External entity loading enabled in parser? • Files read or services hit? • Cloud metadata accessed? • Persistence/post-XXE actions?	• Patch parser (disable external entity processing) • Audit accessed data • Tighten WAF • Egress restriction	TP = XXE + file read or callback. BP = blocked. FP = legitimate XML with external schema.	Patch parser. Disable external entities. Egress lockdown.	Web logs. App logs. File access logs. Outbound from app tier.
61	**Insecure Deserialization**	**High**	Attacker provides malicious serialised object to achieve code execution.	• WAF alert for known deserialization payload (Java, .NET, Python pickle)? • App accepts serialised input from untrusted source? • Process execution from app tier? • Known gadget chain signatures?	• Successful RCE confirmed? • What did the payload do? • Vulnerable library in use (e.g. ysoserial)? • Persistence?	• Patch (avoid native deserialization, validate input) • Reimage app server if RCE • Hunt for persistence • Audit data access	TP = deserialization + RCE. BP = WAF blocked. FP = legitimate serialised content.	Patch. Reimage. Egress lockdown. Reset creds.	Web/app logs. EDR on app server. Payload extraction. RCE evidence.
62	**Path Traversal / LFI**	**High**	Attacker uses ../../ to read files outside the intended directory.	• WAF alert for path traversal patterns? • Sensitive files in web logs (etc/passwd, web.config)? • HTTP 200 responses to traversal payloads? • Source IP making many traversal attempts?	• Which files exposed? • Credentials or keys in those files? • Successful data extraction? • WAF bypass techniques (URL encoding, double encoding)?	• Patch endpoint (canonicalise paths, allow-list) • Rotate any leaked secrets • Audit affected data • Tighten WAF	TP = traversal + sensitive file read. BP = WAF blocked. FP = legitimate path with relative components.	Patch. Block IP. Rotate secrets. WAF tighten.	Web logs. App logs. Files accessed. WAF logs.
63	**Remote File Inclusion (RFI)**	**High**	Attacker makes app load and execute remote file via vulnerable include.	• WAF alert for URL in include parameter? • App tier outbound to external host triggered by request? • Code execution post-RFI? • Source IP enumeration of	• Remote file content (web shell)? • Successful RCE? • Persistence? • Egress to attacker server confirmed?	• Patch (disable allow_url_include or equivalent) • Reimage if RCE • Hunt for web shells • Egress restriction	TP = RFI + remote shell loaded. BP = blocked. FP = legit external resource (rare and risky).	Patch. Reimage. Egress restriction.	Web/app logs. Outbound logs. Remote payload analysis. Web shell hunt.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				vulnerable endpoints?					
64	**Web Shell Upload**	**Critical**	Attacker uploads script (PHP, ASPX, JSP) that gives them RCE on the server.	• File upload endpoint hit + suspicious file extension? • New file in web root with .php/.aspx/.jsp not matching dev pipeline? • Suspicious commands run via web request? • HTTP request triggering process spawn from web user?	• Web shell content (China Chopper, AntSword, Behinder)? • Persistence and lateral movement? • Credentials harvested? • Multiple shells dropped?	• Identify all web shells (file integrity scan) • Reimage or sanitise web server • Patch upload vulnerability • Hunt for credentials taken	TP = web shell file + execution. BP = upload blocked. FP = developer test file (should not be in prod).	Remove shells. Reimage. Patch upload. Reset all related creds.	Web logs. File system snapshots. Process activity from web user. EDR. Web shell sample.
65	**Authentication Bypass**	**Critical**	Attacker accesses protected resources without valid credentials (broken auth, IDOR).	• Access to admin/protected endpoints without auth event? • Direct object reference in URL (?id=123) being incremented? • Session token of one user used by another? • JWT signature manipulation in logs?	• Vulnerability class (IDOR, broken access, JWT tamper)? • Data accessed? • Privileged actions taken? • Pattern of enumeration?	• Patch auth/access logic • Audit data accessed • Reset affected user accounts • Tighten WAF	TP = unauthorised access + actions. BP = blocked. FP = legitimate admin or shared resource.	Patch. Reset accounts. Audit access. WAF tighten.	Web/app logs. Access patterns. JWT analysis. Database access logs.
66	**JWT Token Manipulation**	**High**	Attacker tampers with JWT (algorithm, claims, signature) to gain access.	• JWT with alg=none in logs? • Algorithm switched (RS256 to HS256)? • Anomalous claims (admin: true)? • Tokens with mismatched signatures?	• Did manipulated token grant access? • Library vulnerability used? • Token replay across users? • Privileged actions taken?	• Patch JWT validation (strict alg check, signature verify) • Rotate signing keys • Reset affected sessions • Audit access	TP = manipulated JWT accepted. BP = rejected by validation. FP = legit token rotation.	Patch. Rotate keys. Invalidate active tokens. Reset.	App/auth logs. JWT samples. Access patterns. Library version.
67	**Host Header Injection**	**Medium**	Manipulating HTTP Host header to redirect, poison cache, or reach unintended backends.	• Anomalous Host header values in logs? • Cache poisoning indicators? • Password reset emails with attacker-controlled domain? • Internal hostnames in Host header?	• Was attacker-controlled link sent in real emails (password reset poisoning)? • Cache served wrong content? • Backend routing affected? • Multiple users impacted?	• Patch (validate Host header, use canonical URL) • Invalidate poisoned cache • Notify affected users (especially password resets) • WAF tighten	TP = Host header abuse + impact. BP = blocked. FP = legit multi-site setup with header variation.	Patch. Cache invalidate. Notify users.	Web logs (Host header field). Cache contents. Email logs (password reset traces).

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
68	**Open Redirect**	**Low**	Vulnerable redirect parameter sends users to attacker-controlled URL.	• URL parameter accepting external URLs in redirect? • User reports being sent to phishing page after legitimate site click? • Phishing campaign abusing your redirect? • Multiple users redirected via your site?	• Attacker abusing for phishing trust transfer? • Open redirect chained with auth (OAuth)? • Redirect URLs in logs? • Volume of abuse?	• Patch (allow-list redirect destinations) • Notify affected users • Hunt for phishing using redirect • WAF tighten	TP = redirect to malicious. BP = blocked by allow-list. FP = legitimate redirect.	Patch with allow-list. Notify users. Update WAF.	Web logs (redirect parameters). User reports. Phishing campaigns leveraging.
69	**API Abuse / Rate-Limit Bypass**	**Medium**	Attacker abuses API for scraping, enumeration, or brute force at scale.	• High request volume from one source/key? • Sequential ID enumeration? • Many failed auth attempts via API? • Distributed sources hitting same endpoint?	• Rate limits bypassed (different keys, distributed)? • Data scraped? • Vulnerable endpoint for enumeration? • Credential stuffing via API?	• Tighten rate limits + WAF • Rotate exposed API keys • Add bot detection • Audit data scraped	TP = abuse + scraping/brute. BP = rate limit caught it. FP = legitimate high-volume client.	Block source. Rotate keys. Rate limit tighten.	API gateway logs. Auth logs. Source attribution. Data accessed.

Cloud & SaaS (12)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
70	**Misconfigured Storage Bucket**	**High**	S3/Azure Blob/GCS bucket exposed publicly, leaking data.	• Bucket marked public in cloud audit? • External access logs to storage? • Sensitive data in exposed bucket? • Surface scanner alerted?	• What data was accessible (PII, secrets, code)? • Access logs showing scrapers or specific actors? • How long was it exposed? • Other buckets similarly exposed?	• Lock bucket immediately • Identify exposure window and accessed data • Notify affected parties / regulators • Implement bucket policy enforcement	TP = public bucket + access by external. BP = caught before exposure. FP = intentionally public (CDN, website).	Restrict bucket policy. Audit data accessed. Cloud security policy enforcement.	Cloud audit (bucket policy changes). Access logs. Data inventory. Configuration history.
71	**IAM Key Compromise**	**Critical**	AWS access key, Azure service principal, or GCP key leaked or stolen.	• Key used from unusual region/IP? • High-volume API calls inconsistent with normal use? • Key found in public repo (GitHub leak alert)? • Attacker actions (instance creation, IAM changes)?	• What did attacker do (mining, exfil, persistence)? • IAM policy changes (privilege escalation)? • Other keys/accounts created? • Cost spike or unusual resource creation?	• Rotate the compromised key immediately • Revoke all sessions • Audit IAM changes and roll back • Cost analysis and resource cleanup	TP = key abuse + actions. BP = key found and rotated before use. FP = legitimate automation.	Rotate key. Revoke sessions. Roll back IAM changes. Audit resources.	CloudTrail/Activity logs. IAM change history. Resource creation logs. Cost anomalies. Public repo leak source.
72	**Cloud Privilege Escalation**	**Critical**	Attacker abuses IAM policies (PassRole, AssumeRole, IAM:CreateUser) to escalate.	• IAM API calls from unusual identity? • PassRole / AssumeRole to higher-privilege role? • CreateUser, CreateAccessKey, AttachUserPolicy events? • Path-based escalation pattern?	• Final privilege achieved? • Resources created or accessed using escalated privilege? • Policy abuse pattern (e.g. iam:PassRole + ec2:RunInstances)? • Persistence created?	• Revoke access, remove rogue users/keys • Audit IAM policies for escalation paths • Implement least-privilege • Hunt for persistence	TP = escalation + abuse. BP = blocked by SCP/policy. FP = legitimate admin action.	Revoke. Roll back IAM. Tighten policies. Hunt persistence.	CloudTrail/Activity. IAM change history. Resource access. Persistence artefacts.
73	**Cloud Resource Hijacking (Cryptomining)**	**High**	Attacker spins up expensive compute in your cloud account for mining.	• Unexpected EC2/VM creation events? • Large instance types in unusual regions? • Cost spike alert? • Outbound connections to mining pools?	• IAM identity that created resources? • Persistence mechanism? • Other resources affected? • Initial access (key compromise, exposed metadata)?	• Terminate hijacked resources • Rotate compromised keys • Audit IAM and resources • File for cloud provider credit if applicable	TP = unauthorised resource + mining. BP = blocked by SCP. FP = legitimate workload misclassified.	Terminate. Rotate keys. Audit. Cost recovery.	CloudTrail/Activity. Resource creation. IAM identity. Mining pool destinations. Cost reports.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
74	**OAuth App Abuse (M365/Workspace)**	**High**	Malicious or over-permissioned third-party app granted access to mailboxes/files.	• Newly registered app in tenant audit? • App with excessive permissions (mail.read, files.read.all)? • User consent or admin consent event? • Unverified publisher?	• Data access patterns by the app? • Multiple users granted same app? • App publisher reputation? • Refresh tokens issued?	• Revoke consent across tenant • Block app ID • Audit data accessed • Restrict user-consent policy	TP = malicious app + data access. BP = consent denied. FP = legitimate business app.	Revoke. Block app. Restrict consent policy. Audit access.	Tenant audit (consent events). App API call logs. Data accessed. User sessions.
75	**Mailbox Forwarding Rule Abuse**	**High**	Attacker creates rule to forward mail externally after compromising account.	• New inbox rule with external forward? • Rule hidden (auto-move to RSS, junk)? • Created from unusual location/IP? • User unaware of rule?	• Where mail is forwarded? • Rule created via OWA, EWS, or Graph API? • Other affected mailboxes? • Tied to BEC campaign?	• Remove rule • Reset password and revoke sessions • Hunt for similar rules across tenant • Block external forwarding by policy	TP = external forward + compromise indicators. BP = rule blocked by policy. FP = legitimate forwarding (rare in enterprise).	Remove rule. Reset cred. Revoke sessions. Block external forwarding tenant-wide.	Mailbox audit (rule creation). Sign-in logs. Forwarded mail content. Source IP.
76	**Conditional Access Bypass**	**High**	Attacker finds gap in conditional access policies (legacy auth, service account, exclusion).	• Successful login that should have required MFA but didn't? • Legacy auth (basic, IMAP, POP) succeeding? • Service account used interactively? • Account excluded from CA policy?	• Which CA gap exploited? • Multiple accounts using same gap? • Geo or device pattern? • Token or session issued?	• Close the gap (block legacy auth, narrow exclusions) • Reset affected accounts • Audit CA policy comprehensively • Force MFA review	TP = bypass + access. BP = caught and blocked. FP = legitimate exception not properly excluded.	Close gap. Reset. Tighten CA. Block legacy.	Sign-in logs. CA policy evaluation logs. Access patterns. Affected accounts.
77	**Cloud Logging Tamper / Disable**	**Critical**	Attacker disables CloudTrail, Activity Log, or Audit Log to hide their tracks.	• StopLogging, DeleteTrail, or audit-disable event? • Gap in logging timeline? • IAM identity that did it? • Followed by other suspicious activity?	• Was logging restored? • What happened in the gap (if forensically reconstructable)? • Multiple regions affected? • Persistence created during gap?	• Re-enable logging • Investigate gap activity from other sources (network, billing) • Treat as advanced compromise • Tighten controls (deny logging-disable in SCP)	TP = logging disabled by attacker. BP = blocked by SCP. FP = legitimate maintenance.	Re-enable logging. SCP enforcement. Treat as breach. Hunt.	Cloud audit. Other data sources for gap reconstruction (NetFlow, billing, snapshots). IAM identity.
78	**Cloud Snapshot / AMI Exfiltration**	**High**	Attacker copies snapshot/disk image and	• CreateSnapshot/CopySnapshot/ModifySnapshotAttribute events?	• What data was on snapshot?	• Revoke snapshot sharing	TP = snapshot shared externally. BP =	Revoke share. SCP.	CloudTrail/Activity. Snapshot history. External

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
			exfiltrates to attacker account.	• Snapshot shared to external account? • AMI made public? • Initiated by unusual identity?	• External account ID — known attacker? • Persistence created? • Other snapshots similarly exposed?	• Audit data exposure • Rotate any creds in snapshot • Notify affected parties if data left org	blocked by SCP. FP = legit cross-account share.	Rotate creds. Notify.	account ID. Data inventory.
79	**Container Escape / Kubernetes Compromise**	**Critical**	Attacker escapes container to host or compromises K8s cluster control plane.	• Privileged container creation events? • Anomalous kubectl exec? • Service account token abuse? • Pod with hostPath, hostNetwork, or capabilities?	• Cluster admin creds compromised? • Workload identity abuse? • Persistence (DaemonSet, CronJob)? • Image registry compromise?	• Rotate cluster certs and tokens • Audit RBAC • Reimage cluster nodes if escape confirmed • Image registry hardening	TP = escape + host control. BP = blocked by PSS / admission controller. FP = privileged DevOps action.	Rotate. Reimage nodes. RBAC tighten. Admission controller.	K8s audit logs. Container runtime logs. Image registry logs. Host EDR.
80	**Serverless Function Abuse**	**Medium**	Attacker abuses Lambda/Functions for cryptomining, persistence, or pivot.	• New unusual functions in account? • Functions invoking from unexpected source? • High invocation count or duration? • Outbound to mining pool from function?	• Function code reviewed (mining script, backdoor)? • IAM role attached and over-permissioned? • Persistence via function trigger (S3 event, schedule)? • Cost spike?	• Delete malicious functions • Audit IAM roles for serverless • Rotate any creds in env vars • Restrict function deployment	TP = malicious function deployed. BP = blocked by SCP. FP = legitimate dev test function.	Delete. Rotate creds. Restrict deployment.	Function logs. Deployment history. IAM identity. Cost. Function source code.
81	**SaaS App Token Theft**	**High**	Attacker steals SaaS API tokens (Slack, GitHub, Salesforce) for persistent access.	• Token usage from unusual IP? • API calls outside business hours? • Mass data export via API? • Token issued via OAuth from suspicious app?	• What was accessed (channels, repos, records)? • Token scope and lifetime? • Tied to broader phishing or breach? • Multiple SaaS apps affected?	• Revoke tokens • Reset user creds • Audit data accessed • Implement IP allow-listing on SaaS where possible	TP = token abuse from unexpected source. BP = caught by SaaS anomaly detection. FP = developer travel/automation.	Revoke. Reset. IP allow-list. Audit.	SaaS audit logs. Token scope and usage. Source IP. Data accessed.

Active Directory & Kerberos (10)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
82	**Group Membership Modification (Privilege Add)**	**High**	User added to a privileged group (Domain Admins, Enterprise Admins) — can be insider or attacker post-compromise.	• 4728/4732/4756 events for privileged group changes? • Who added whom and when? • Modifying user is a regular admin or unexpected? • Target user role justifies the privilege?	• Was the addition pre-approved (change ticket)? • Source workstation of the change? • Target user account compromise indicators? • Multiple recent privilege changes?	• Reverse the change if unauthorised • Investigate source admin account • Reset both source and target credentials if compromise • Tighten privileged group monitoring	TP = unauthorised privileged group add. BP = revert before exploitation. FP = approved change with ticket.	Remove from group. Reset both accounts. Investigate source.	4728/4732/4756 events. Source workstation logs. Change-management records. Account activity post-change.
83	**AdminSDHolder Abuse**	**High**	Attacker modifies AdminSDHolder permissions to maintain admin rights persistently.	• Permission changes on AdminSDHolder object? • ACL modifications affecting protected groups? • SDProp running unexpectedly? • Anomalous permissions on Domain Admins?	• Persistence pattern (changes survive group removal)? • Multiple objects affected? • Source compromise of admin account? • Tools (PowerSploit) traces?	• Restore AdminSDHolder ACLs • Reset all admin accounts • Audit protected groups • Tier-0 review	TP = unauthorised AdminSDHolder change. BP = blocked. FP = sanctioned IT permission change.	Restore ACLs. Reset admins. Tier-0 lockdown.	4670 events on AdminSDHolder. ACL change history. Source endpoint. AD permission audit.
84	**Group Policy Object (GPO) Abuse**	**High**	Attacker creates/modifies GPO to deploy malware, scripts, or backdoors at scale.	• GPO created or modified by non-standard admin? • GPO contains script (logon, startup)? • GPO links to OUs with sensitive systems? • SYSVOL changes in scripts folder?	• GPO content (PowerShell, scheduled task, file deployment)? • Targets and applied users/computers? • Source workstation of admin action? • Persistence intent?	• Revert GPO changes • Hunt for hosts that already applied the GPO • Reset admin creds • Audit GPO permissions	TP = malicious GPO deployed. BP = caught before apply. FP = legitimate GPO change.	Revert. Reset admins. Hunt applied targets.	GPO change events. SYSVOL contents. Endpoints applied. Source admin activity.
85	**DC Shadow Attack**	**Critical**	Attacker registers rogue DC to inject malicious changes that replicate to real DCs.	• New DC registration in domain? • Non-DC source initiating replication? • SPN changes for HOST/GC services? • DRSReplicaAdd events from unusual source?	• Confirmed forged DC? • What changes injected? • Persistence via injected modifications? • Mimikatz lsadump::dcshadow signatures?	• Treat as full AD compromise • KRBTGT double-reset • Forensic preserve all DCs • Rebuild trust posture	TP = rogue DC + replication. BP = blocked. FP = legitimate DC promotion (rare).	Full AD compromise response. KRBTGT reset twice. Image DCs.	DC logs (replication events). SPN changes. Endpoint EDR for Mimikatz. AD object changes.
86	**NTLM Relay Attack**	**High**	Attacker relays NTLM auth from one host to another to authenticate as victim.	• NTLM auth from unusual source to high-value target? • Victim account auth where it shouldn't be?	• Source endpoint compromise? • Target service (LDAP, MSSQL, ADCS)? • Did relay succeed?	• Enable SMB signing, LDAP signing, EPA • Disable NTLM where possible	TP = relay + access. BP = signing prevented. FP = legitimate NTLM auth.	Enable signing. Disable NTLM. Reset accounts.	4624 events. NTLM auth chain. Network captures. ADCS logs (if applicable).

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• Tool indicators (Responder, ntlmrelayx)? • SMB / LDAP / HTTP relay patterns?	• Persistence created (ADCS cert request)?	• Reset relayed accounts • ADCS hardening if certs issued			
87	**PetitPotam / Coerced Authentication**	**High**	Attacker forces a target (often DC) to authenticate to attacker, then relays.	• EFSRPC / DFSCoerce / PrinterBug call to DC? • Outbound auth from DC to unexpected destination? • Followed by NTLM relay attempt? • Tools (PetitPotam, Coercer) indicators?	• Target service of relay (often ADCS)? • Cert issued via relay? • Persistence (NTAuth cert)? • Other DCs targeted?	• Patch (KB5005413, ADCS hardening) • Disable NTLM authentication for ADCS • Revoke any improper certs • Hunt for cert-based persistence	TP = coercion + relay + cert. BP = blocked by patch. FP = none reasonable.	Patch. Disable NTLM on ADCS. Revoke certs.	DC RPC logs. ADCS issuance logs. Network captures. Cert audit.
88	**Certificate Services (ADCS) Abuse**	**Critical**	Attacker exploits ADCS misconfig (ESC1-ESC8) to issue certs and impersonate users.	• Cert request for SAN of another user? • Vulnerable template enabled (ESC1: enrollee supplies subject)? • Cert issued to non-admin requestor for admin SAN? • Tools (Certify, Certipy) indicators?	• Which ESC technique? • Cert used to authenticate as victim? • Persistence (long-lived cert)? • Domain admin impersonation?	• Disable vulnerable templates • Revoke all suspect certs • Reset impacted accounts (twice for KRBTGT-class) • ADCS hardening (Locksmith, etc)	TP = bad template abused + cert auth. BP = template hardened. FP = legitimate cert request.	Disable templates. Revoke certs. Reset accounts. Harden.	ADCS issuance logs. Cert details. Auth events using cert. Endpoint EDR.
89	**Skeleton Key Attack**	**Critical**	Attacker patches LSASS on DC with master password — any user authenticates with that password.	• LSASS process patched (in-memory)? • All accounts succeeding with one constant password? • Mimikatz misc::skeleton signatures? • DC compromise indicators?	• Persistence — survives reboot? • Active hands-on attacker? • Other DCs affected? • Lateral movement using skeleton key?	• Reboot DC (skeleton key clears unless persisted) • Full AD compromise response • KRBTGT reset twice • Forensic image of DC	TP = LSASS patch + universal-password auth. BP = LSASS access blocked. FP = none.	Reboot DC. AD compromise response. Image DCs.	DC memory image. LSASS access events. Auth pattern analysis. EDR on DC.
90	**Trust Abuse / Cross-Domain Attack**	**High**	Attacker abuses domain trust to move from compromised domain to trusting domain.	• Cross-domain auth from compromised domain? • SID filtering bypass attempts? • TGT with foreign SID in PAC? • TrustedDomain enumeration?	• Trust direction and type (transitive, external, forest)? • SID History abused? • Mimikatz golden ticket with cross-domain SID? • Privileged access in trusting domain?	• Enable SID filtering • Audit trusts (remove unneeded) • Reset both KRBTGTs (twice) • Tier-0 review across forest	TP = cross-domain abuse. BP = SID filtering caught it. FP = legitimate cross-domain admin.	SID filtering. KRBTGT reset both domains. Audit trusts.	DC logs in both domains. Kerberos events. SID History audit. Trust configuration.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
91	**GPP Password Disclosure**	**Medium**	Group Policy Preferences (cpassword) in SYSVOL contains decryptable passwords.	• cpassword strings in SYSVOL XML files? • SYSVOL access by non-admin user? • Tool indicators (Get-GPPPassword)? • Recent SYSVOL enumeration?	• Which accounts had passwords stored? • Accounts still active with same password? • Lateral movement using disclosed creds? • SYSVOL access pattern?	• Remove all cpassword from SYSVOL • Reset all accounts with disclosed passwords • Tighten SYSVOL access • Brief on GPP best practices	TP = cpassword found and used. BP = found before use. FP = legitimate access scoped properly.	Remove cpassword. Reset accounts. Tighten SYSVOL.	SYSVOL file audit. Account password history. SYSVOL access logs.

Insider & Data Exfiltration (8)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
92	**Mass File Download (SharePoint / OneDrive / Drive)**	**High**	User downloads unusual volume of files from corporate file storage.	• DLP / CASB alert for high-volume download? • User context — leaving role, recent perf issues? • Files sensitive (PII, IP, contracts)? • Off-hours or unusual location?	• Files downloaded — what categories? • Pattern (bulk download tool, manual)? • Sync to personal device? • Account compromise vs insider intent?	• Confirm intent with user/manager • Legal hold on user's data • If insider: HR/Legal coordination • If compromise: full IR	TP = mass download + sensitive data + intent unclear. BP = caught early, contained. FP = legitimate bulk work.	Disable user account temporarily. Legal hold. Forensic preserve.	DLP/CASB logs. SharePoint/OneDrive audit. Device sync logs. User communication review.
93	**USB / Removable Media Exfiltration**	**Medium**	Sensitive data copied to USB drive.	• USB device insertion + large file copy? • User profile (departing, disgruntled)? • Files copied — sensitive? • Frequency of USB use?	• Device serial number — personal or corporate? • Encryption applied (BitLocker)? • Repeat behaviour over time? • Cross-correlate with email/cloud exfil?	• HR/Legal coordination if insider • Block USB on critical data systems • Recover device if possible • Update DLP rules	TP = sensitive data + personal USB + intent. BP = blocked by DLP. FP = sanctioned offline transfer.	Block USB write. HR/Legal. Recover device.	USB device logs. File copy events. EDR timeline. Device serial. Recovered device image.
94	**Email-Based Exfiltration**	**Medium**	User emails sensitive data to personal address or external party.	• DLP alert for sensitive content sent externally? • Recipient is personal email (gmail, yahoo)? • Multiple emails to same external? • Attachment patterns (zip, encrypted)?	• Pattern over time (just-leaving behaviour)? • Encrypted attachments to evade DLP? • Volume and content? • Coordinated with download spike?	• HR/Legal • Quarantine future emails • Recover sent content if possible • Tighten DLP	TP = sensitive + external personal + pattern. BP = DLP blocked. FP = legitimate external sharing.	HR/Legal. DLP tighten. Quarantine future.	Email audit (sent items). DLP logs. Recipient list. Content review.
95	**Cloud Storage Exfiltration (Dropbox, Box, personal Drive)**	**Medium**	User uploads sensitive data to personal cloud storage.	• Outbound to personal cloud service from corporate device? • CASB alert? • Volume of upload? • User profile?	• Same user using corporate cloud less? • Other channels (USB, email) also active? • Coordinated exfil pattern? • Browser-based vs client-app upload?	• HR/Legal • Block personal cloud at proxy • CASB shadow IT review • Tighten policy	TP = personal cloud + sensitive content + pattern. BP = blocked. FP = sanctioned use.	Block at proxy. HR/Legal.	Proxy/CASB logs. Endpoint EDR. Personal cloud usage history. User communications.
96	**Source Code Theft**	**High**	Developer pushing code to personal repo or downloading entire repo before leaving.	• GitHub/GitLab clone of entire repos? • Push to personal/public repo from corporate identity? • User in dev role and departing?	• Sensitive repos (proprietary algorithms, secrets)? • Same user accessing repos they don't normally?	• Legal hold and HR • Audit external pushes • Revoke user access to repos • Remove any pushed content if possible	TP = unauthorised external push + sensitive code + departing. BP = blocked. FP = sanctioned open-source contribution.	Revoke access. Legal hold. Remove external content.	Git logs (clones, pushes). External repo history. User device EDR. Email/cloud cross-check.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• Volume of code transferred?	• Other channels (USB, email) coordinated? • Intent indicators?				
97	**Print-Based Exfiltration**	**Low**	Sensitive documents printed for physical removal.	• Print volume spike for one user? • Printing sensitive documents? • Print-server logs showing pattern? • User departing or under investigation?	• Documents printed — what? • Time of printing (after hours)? • Other exfil channels active? • Frequency change over time?	• HR/Legal • Restrict print access • Pull-print review • Brief manager	TP = sensitive + volume + intent. BP = blocked by DLP-print. FP = legitimate need.	Restrict print. HR/Legal.	Print server logs. Document content. Pull-print history. User access patterns.
98	**Database Mass Export**	**High**	Insider runs unusually large query or export from production database.	• DB audit alert for large result set? • User querying tables they don't normally? • Export to file or another system? • Query outside normal hours?	• Volume and content (PII, financial, IP)? • Repeated pattern? • User role and departure status? • Cross-correlate with other exfil signals?	• HR/Legal • Restrict DB access • Audit exported data • Tighten DB DLP / row-level security	TP = mass export + sensitive + intent. BP = blocked by row-limits. FP = legitimate analytics.	Restrict DB. HR/Legal. Audit export.	DB audit logs. Query history. Export destination. User profile and activity.
99	**Privileged Access Abuse (Sysadmin reading mail/docs)**	**Medium**	IT admin accesses user mailboxes, files, or systems beyond their job need.	• Admin accessing user mailbox without ticket? • Admin reading executive emails? • Pattern of access without service request? • Snooping behaviour over time?	• Records of access vs tickets? • User complaints? • Sensitive data accessed? • Other admins doing similar?	• HR/Legal • Tighten privileged access (PIM, JIT) • Audit all admin access historically • Brief on policy	TP = unsanctioned privileged access. BP = blocked by PIM. FP = legitimate troubleshooting.	Revoke standing privilege. PIM/JIT. HR.	Admin audit logs. Access requests vs tickets. Specific data accessed. User complaints.

OT / ICS / IoT (8)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
100	**PLC Manipulation (Stuxnet-class)**	**Critical**	Attacker modifies programmable logic controller logic to cause physical effects.	• Anomalous engineering workstation activity? • PLC programming change events? • Process value deviations? • Operator reports of unexpected behaviour?	• Source of programming change (auth and timing)? • Logic modification details? • Other PLCs affected? • Air-gap breach evidence?	• Stop affected process safely • Revert PLC logic from known-good baseline • Forensic image PLCs and engineering workstations • Engage ICS specialist	TP = unauthorised PLC mod + impact. BP = caught at change-control. FP = legitimate engineering update.	Stop process safely. Revert logic. Isolate OT segment.	PLC logic snapshots. Engineering workstation logs. Process historian. OT-aware monitoring.
101	**ICS Protocol Abuse (Modbus, DNP3, S7)**	**High**	Attacker sends malicious ICS protocol commands to PLCs or RTUs.	• Modbus/DNP3/S7 commands from non-engineering source? • Function codes for write/control from unusual host? • Anomalous register changes? • OT IDS (Claroty, Nozomi) alert?	• Source — IT-OT bridge compromise? • Specific function codes used (start/stop)? • Process impact? • Other devices targeted?	• Block source at IT-OT firewall • Audit OT segmentation • Engage ICS team • Forensic preserve	TP = unauthorised control commands. BP = blocked at OT firewall. FP = legitimate engineering tool.	Block source. OT segmentation review. Engage ICS team.	OT IDS logs. Firewall logs. PLC logs. Engineering workstation EDR.
102	**IT-OT Pivot Attack**	**Critical**	Attacker compromises IT network, then pivots through DMZ into OT.	• Connections from IT to OT through DMZ? • IT host showing OT-specific tools or queries? • DMZ jump host compromise indicators? • Engineering laptop dual-homed?	• Path of pivot (specific hosts/ports)? • Dwell time before OT engagement? • OT impact yet? • Air-gap supposed but breached?	• Sever IT-OT connection at firewall • Engage ICS team and IR • Reimage all suspect hosts • OT segmentation hardening	TP = IT-to-OT pivot. BP = blocked at DMZ. FP = sanctioned IT-OT data flow.	Sever IT-OT. Reimage. ICS team engage.	IT EDR. DMZ firewall + jump host logs. OT IDS. Process historian.
103	**HMI Compromise**	**High**	Attacker compromises Human-Machine Interface to issue malicious commands or hide alarms.	• HMI showing inconsistent values vs PLC ground truth? • HMI logs showing commands operator denies sending? • Alarm suppression on HMI? • RDP/remote access to HMI from unusual source?	• Account used (operator or admin)? • Persistence on HMI? • Process impact? • Operator misled by display?	• Restore HMI from baseline • Cross-validate with PLC and historian • Revoke remote access to HMI • ICS team review	TP = HMI compromise + manipulation. BP = caught early. FP = legitimate operator action.	Restore. Sever remote access. ICS team.	HMI logs. PLC + historian cross-check. Remote access logs. Operator interview.
104	**IoT Botnet Recruitment (Mirai-class)**	**Medium**	IoT devices (cameras, routers, DVRs) compromised and added to botnet.	• IoT device outbound to known botnet C2? • Default credentials accessed externally? • Telnet/SSH brute force from internet to IoT?	• Number of devices affected? • Botnet family (Mirai variant)? • Internal pivot from IoT to corporate? • ISP or vendor notification?	• Reset/reflash devices • Change default creds • Segment IoT to its own VLAN • Block C2	TP = device + botnet activity. BP = device hardened before infection. FP = legitimate vendor cloud connection.	Reflash. Segment. Block C2. Reset creds.	Network connections from IoT. Device firmware. Botnet C2 IOCs.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• Devices participating in DDoS?					ISP/vendor reports.
105	**Building Management System (BMS) Compromise**	**High**	Attacker compromises HVAC, lighting, access control, or facility management.	• BMS portal access from unusual source? • BMS commands setting extreme parameters (temp, lighting)? • Physical access control changes (door unlocks)? • Vendor remote access misuse?	• Operational impact? • Vendor account or local account? • Persistence? • Linked to physical security event?	• Restore BMS to baseline • Revoke vendor access • Coordinate with facilities team • Engage physical security	TP = unauthorised BMS control. BP = caught early. FP = sanctioned vendor maintenance.	Restore. Revoke. Physical security loop.	BMS logs. Physical access logs. Vendor remote-session logs. Facility CCTV.
106	**OT Pre-positioning (Volt Typhoon-style)**	**Critical**	Long-dwell access in critical infrastructure for use during future conflict.	• LOLBin patterns on OT-adjacent hosts? • Long-dwell credential reuse? • Living-off-the-land in DMZ? • Persistence with no immediate impact?	• Attribution to known nation-state TTP? • Dwell time and access scope? • Critical-control reach? • Coordination with intel agencies (CISA-style)?	• Treat as nation-state IR • Engage CISA/govt CERT • Forensic preserve everything • Long-term hunt and remediation	TP = long-dwell + nation-state TTP + critical access. BP = caught early. FP = none reasonable.	Engage govt CERT. Tier-0 lockdown. Long-term hunt.	All logs maximum retention. Memory + disk images. EDR full timeline. Threat intel coordination.
107	**Safety Instrumented System (SIS) Tampering**	**Critical**	Attacker disables safety controls (Triton/TRISIS-style) — risk of physical harm.	• SIS configuration change events? • Engineering workstation accessing SIS? • Anomalous logic uploads to SIS controllers? • TRITON/TRISIS-class tool indicators?	• Safety logic modified? • Process state — running with modified SIS? • Air-gap breach? • Vendor or insider involvement?	• IMMEDIATE — coordinate safe shutdown • Restore SIS from known-good • Engage ICS specialist + vendor • Notify regulators if applicable	TP = SIS tamper confirmed. BP = blocked. FP = sanctioned engineering update.	Safe shutdown. Restore. Engage specialists.	SIS logic snapshots. Engineering workstation EDR. Process historian. Vendor coordination.

Supply Chain (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
108	**Software Supply Chain Compromise**	**Critical**	Malicious code introduced into a legitimate software product (SolarWinds, 3CX-style).	• Vendor advisory of compromise? • Detection on hosts running affected software? • Outbound to attacker C2 from vendor process? • Hash/signature anomaly in vendor binary?	• Affected version range? • Persistence and post-compromise activity? • Other affected hosts in estate? • Dwell time before disclosure?	• Patch or remove affected software • Reimage hosts where compromise confirmed • Hunt across estate • Coordinate with vendor and threat intel	TP = vendor confirmed + IOCs match. BP = caught at update review. FP = false vendor alert.	Patch/remove. Reimage. Hunt. Vendor coordination.	Vendor IOCs. EDR timeline on affected hosts. Network captures. Persistence artefacts.
109	**Malicious npm/PyPI/Package**	**High**	Attacker publishes malicious package or compromises real one (typosquatting, dependency confusion).	• Build/dev pipeline using new or odd-named package? • Outbound from build server during install? • Package recently published or version bumped suddenly? • Known typosquat warning from registry?	• Package contents (postinstall script, exfil)? • Other repos using same package? • Build server compromise? • Secrets in env vars exfiltrated?	• Remove package from build pipelines • Rotate any secrets exposed • Image build server if compromised • Block package and similar names	TP = malicious package + execution. BP = caught at SCA. FP = legitimate new package.	Remove. Rotate secrets. Image build. SCA.	Build logs. Package source. Network from build server. Secret exposure check.
110	**Vendor / MSP Compromise**	**Critical**	MSP, IT vendor, or SaaS provider with privileged access is breached and reused for downstream.	• Vendor remote access from unusual source? • Vendor account doing things outside scope? • Vendor's other clients seeing similar activity (if disclosed)? • Vendor advisory of breach?	• Scope of vendor access in your env? • What did attacker do? • Persistence beyond vendor session? • Data accessed?	• Cut vendor access • Reset all creds vendor had • Hunt for persistence • Coordinate with vendor and other affected	TP = vendor account abused. BP = vendor access caught early. FP = sanctioned vendor work.	Cut access. Reset creds. Hunt. Vendor coordinate.	Vendor remote access logs. Actions taken via vendor account. Persistence artefacts. Data access.
111	**CI/CD Pipeline Compromise**	**Critical**	Attacker compromises build pipeline (Jenkins, GitHub Actions, Azure DevOps) to inject malicious code into releases.	• Anomalous pipeline runs? • New pipeline secret or service account? • Build artifacts with unexpected changes? • External callouts from pipeline?	• Source — token theft, malicious commit, dependency? • Releases tainted shipped to customers? • Persistence in pipeline? • Other pipelines affected?	• Halt pipelines • Rotate all pipeline secrets • Audit recent releases for tampering • Coordinate with downstream customers if released	TP = pipeline compromise + tainted release. BP = caught before release. FP = legitimate config change.	Halt. Rotate. Audit releases. Customer coordinate.	Pipeline logs. Commit history. Secret rotation history. Build artifact comparison.
112	**Hardware Supply Chain (Firmware Implant)**	**Critical**	Malicious firmware or implant introduced during	• Firmware integrity check failures? • Anomalous boot behaviour?	• Affected device population? • Persistence below OS?	• Coordinate with vendor and intel agencies	TP = confirmed implant. BP = caught at receiving. FP =	Quarantine. Replace. Vendor coordinate.	Firmware dumps. Device inspection. Vendor

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
			manufacturing or shipment.	• Unexpected device on supply chain? • Vendor advisory of tampering?	• Vendor or shipping intermediary compromise? • Nation-state attribution?	• Quarantine affected devices • Replace where feasible • Long-term firmware-integrity program	legitimate vendor update.		records. Shipment trail.
113	**Trusted Relationship Abuse (B2B Connector)**	**High**	Attacker uses trust between organisations (federated identity, SAML trust, B2B guest) to pivot.	• Guest user from compromised partner active? • Federated SAML auth from unusual source? • Cross-tenant access events? • Partner organisation breach disclosure?	• Scope of guest access? • Actions taken via federated identity? • Persistence in your tenant? • Other tenants of same partner affected?	• Suspend guest accounts from compromised partner • Tighten cross-tenant access policies • Hunt for actions taken • Coordinate with partner	TP = federated abuse + actions. BP = blocked by CA. FP = legitimate partner work.	Suspend guests. Tighten CA. Coordinate.	Sign-in logs (cross-tenant). Guest activity. Partner breach reports. CA evaluation.

Mobile (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
114	**Mobile Malware (Android/iOS)**	**High**	Malicious app installed on mobile device, often via sideload or malicious store.	• MDM / MTD alert for malicious app? • Sideloaded app on managed device? • App requesting excessive permissions? • User reports phone behaving oddly?	• Malware family (banking trojan, spyware, stalkerware)? • Data accessed (contacts, SMS, mic, camera)? • C2 traffic from device? • Other devices in fleet?	• Wipe device via MDM • Block sideloading • Reset corporate accounts user had on device • Hunt for fleet impact	TP = malware confirmed + data access. BP = blocked by MDM. FP = legitimate app flagged.	Wipe. Reset accounts. MDM lock down sideloading.	MDM logs. App package and behaviour. Network from device. User account access logs.
115	**SIM Swap Attack**	**Critical**	Attacker convinces carrier to transfer victim's number to attacker SIM, intercepting SMS MFA.	• User reports loss of cellular service? • SMS MFA codes not arriving? • Account password reset triggered without user knowledge? • Carrier port-out alert?	• When did port happen? • What accounts used SMS MFA? • Account compromise downstream? • Coordination with carrier and law enforcement?	• Recover number via carrier • Reset all accounts user has • Move to app-based or hardware MFA • File with FBI IC3 / regulator	TP = port + account compromise. BP = port-out caught and reversed quickly. FP = user actually changed carrier.	Recover number. Reset all accounts. Stronger MFA. Carrier port-protection.	Carrier records. Account auth logs. Password resets. Communication records.
116	**Malicious Mobile Profile (iOS MDM/Configuration Profile)**	**High**	User tricked into installing malicious configuration profile that grants attacker control.	• Unmanaged config profile installed? • Profile from suspicious URL? • VPN auto-configured to attacker server? • Cert installed allowing TLS interception?	• What profile capabilities (proxy, cert, VPN, MDM)? • Traffic routing through attacker? • Data accessed? • Other users with same profile?	• Remove profile • Wipe device • Brief users on profile install warnings • Tighten device policy	TP = malicious profile installed. BP = blocked by managed policy. FP = legitimate enterprise profile.	Remove profile. Wipe. Tighten policy.	Device profile logs. Network traffic. Profile package. User interaction.
117	**Mobile Phishing (Smishing follow-up)**	**Medium**	Phishing via SMS, WhatsApp, social media DM leading to credential theft or malware.	• User reports phishing message? • Multiple users got same message? • Link reputation on link in message? • User clicked or interacted?	• If clicked: did they enter creds or install app? • Channel (SMS, WhatsApp, Telegram, Signal)? • Mobile or also desktop fallback? • Pattern matches known campaign?	• Block link at MDM/proxy • Reset user creds if entered • Brief users • Update awareness training	TP = phish + interaction + creds/malware. BP = blocked. FP = legitimate.	Block. Reset. Brief.	Mobile logs. Network. Message content. Account activity post-click.
118	**Pegasus / Commercial Spyware**	**Critical**	State-grade spyware (Pegasus, Predator) on a high-value target's phone.	• Citizen Lab / Amnesty / vendor advisory match? • High-value target (journalist, exec, activist)?	• MVT (Mobile Verification Toolkit) findings? • Iran/NSO/Cytrox attribution patterns? • Persistence mechanism? • Data exfil scope?	• Replace device entirely • Reset all user accounts • Engage specialist incident response • Brief target on personal security	TP = MVT + indicators of state spyware. BP = caught at network level. FP = none reasonable.	Replace device. Reset everything. Specialist IR.	MVT scan. iTunes/iOS backup analysis. Network traffic capture. Specialist coordination.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• Anomalous battery, data usage, processes? • Zero-click delivery indicators?					
119	**Mobile App Token Theft**	**High**	Tokens stored in mobile app extracted via jailbreak, compromised app, or backup.	• Token used from non-mobile source? • Jailbreak detection alert? • Token reuse across IPs? • Mobile app crash or compromise reported?	• Token scope? • What was accessed? • App vulnerability used? • Backup compromise (iCloud/Google)?	• Revoke tokens • Force re-auth on app • Patch app (token storage hardening) • Audit access	TP = token replay from other source. BP = caught by anomaly detection. FP = user multi-device.	Revoke. Patch app. Re-auth.	Token usage logs. App audit. Mobile device forensics. Backup access logs.

DDoS & Availability (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
120	**Volumetric DDoS (Layer 3/4)**	**High**	Massive traffic flood (UDP, SYN, amplification) overwhelming bandwidth.	• Bandwidth saturation alert? • Traffic from many distinct IPs to single target? • Specific protocol (DNS, NTP, memcached) amplified? • Service availability degraded?	• Attack type (SYN flood, UDP flood, amplification)? • Source distribution (botnet, reflectors)? • Target — single service or wider? • Volume in Gbps/Mpps?	• Engage DDoS scrubbing service • Anycast / CDN front • Filter at upstream ISP • Rate-limit at edge	TP = volume + service impact. BP = absorbed by scrubbing. FP = legitimate viral traffic.	Scrubbing. ISP filter. Edge rate-limit.	NetFlow. ISP logs. Source IP geo distribution. Attack vector identification.
121	**Application Layer DDoS (Layer 7)**	**High**	HTTP/HTTPS flood targeting application logic to exhaust resources.	• High request rate to specific endpoint? • Few sources or distributed? • CPU/DB exhaustion on web tier? • Slowloris / RUDY patterns?	• Endpoint targeted (login, search, expensive query)? • User-agent and behaviour patterns? • Bot characteristics? • Application response time degradation?	• WAF rate-limiting and bot management • Cache static endpoints • CAPTCHA on suspicious endpoints • Application-level optimisation	TP = abuse pattern + service impact. BP = WAF caught. FP = legit campaign or product launch.	WAF + bot management. CAPTCHA. Cache.	WAF logs. Web logs. Application performance. Source patterns.
122	**DNS DDoS / NXDOMAIN flood**	**Medium**	Attacker floods DNS resolver with non-existent queries to exhaust resources.	• Spike in DNS queries? • High NXDOMAIN response rate? • Source distribution wide? • Authoritative server affected?	• Attack target — recursive or authoritative? • Source — botnet or specific? • DNSSEC amplification? • Volume?	• Rate-limit DNS • Anycast / DNS provider scrubbing • Block source ASNs • Tighten DNS architecture	TP = DNS abuse + impact. BP = caught by rate limit. FP = legitimate query burst.	Rate-limit. Scrubbing. Block sources.	DNS logs. NXDOMAIN ratios. Source IPs.
123	**DDoS Extortion (Ransom DDoS)**	**High**	Attacker threatens or launches DDoS demanding payment.	• Ransom email/note received before/during attack? • Attack pattern aligns with known group (e.g. Fancy Lazarus)? • Demand for crypto payment? • Initial small attack as proof?	• Attack scale and duration? • Group attribution? • Coordinate with peers / ISACs? • Payment pressure tactics?	• Do not pay • Engage DDoS scrubbing • Notify law enforcement • ISAC / threat-intel sharing	TP = ransom + attack. BP = absorbed. FP = none reasonable.	Scrubbing. LE engage. ISAC.	Attack traffic. Ransom communication. Crypto wallet (for LE). ISAC reports.
124	**Resource Exhaustion (Connection / DB Pool)**	**Medium**	Attacker opens many slow connections to exhaust app pools or DB connections.	• Connection pool exhausted in app? • Many idle/slow connections from few IPs? • DB connections maxed? • Service timeouts to legitimate users?	• Pattern (Slowloris, slow-POST, slow-read)? • Source distribution? • Application config tunable? • Coordinated with other attacks?	• Tighten timeouts • Connection limits per IP • WAF rules for slow attacks • Capacity planning	TP = slow exhaustion + impact. BP = blocked. FP = legit slow client.	Tighten timeouts. WAF. Limits.	Web/app logs. Connection state. Source IPs.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
125	**Botnet DDoS (Mirai-class)**	**High**	Massive DDoS from compromised IoT/edge devices.	• Source distribution wide and includes residential/IoT IPs? • Volume Tbps-class? • Attack patterns matching Mirai variants? • Notable target (provider, large enterprise)?	• Botnet attribution? • C2 known to IPs? • Defenders coordinated (Cloudflare, Akamai reports)? • Volume and protocol mix?	• Scrubbing + ISP coordination • Block C2 globally • ISAC + LE • Defensive infrastructure scaling	TP = botnet + impact. BP = absorbed. FP = none.	Scrubbing. C2 block. ISP/LE coord.	NetFlow. Source attribution. Botnet C2 IOCs. Provider reports.

Wireless & Physical (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
126	**Rogue Access Point / Evil Twin**	**Medium**	Attacker sets up fake Wi-Fi mimicking corporate SSID to capture creds and traffic.	• WIDS alert for unauthorised AP? • SSID matching corporate from unknown BSSID? • Users complaining of disconnections? • RSSI patterns suggesting rogue AP?	• AP MAC vendor (rogue device hardware)? • Captive portal harvesting creds? • Users connected and sharing data? • Physical location of rogue AP?	• Locate and remove rogue AP • Reset creds of users who connected • WIDS/WIPS hardening • Brief users on Wi-Fi practices	TP = rogue + user connection. BP = caught by WIDS. FP = legitimate guest AP.	Remove AP. Reset creds. WIDS.	WIDS logs. AP physical recovery. User connection logs. Captive portal trace.
127	**Wi-Fi WPA Cracking**	**Medium**	Attacker captures WPA handshake and cracks offline to gain network access.	• Deauth attacks observed in WIDS? • Multiple deauth events from one source? • Devices reconnecting frequently? • Capture of EAPOL handshake suspected?	• WPA2-PSK with weak passphrase? • WPA3 in use? • Network access post-crack? • Tools (aircrack, hashcat) traces externally?	• Move to WPA3-Enterprise • Strong passphrases (24+ char) for any PSK • 802.1X for corp networks • Monitor deauth patterns	TP = deauth + suspected crack + access. BP = WPA3 strong. FP = legitimate deauth (rare).	Move WPA3. 802.1X. Strong passphrase.	WIDS/WIPS logs. Captures. Network access logs post-event.
128	**Bluetooth Attacks (BlueBorne, BIAS)**	**Medium**	Attacker exploits Bluetooth vulnerabilities or weak pairing.	• Bluetooth scanner alerts? • Devices in discoverable mode unnecessarily? • Known Bluetooth vuln in fleet? • Unusual pairing events?	• Targeted device value (mobile, peripheral)? • Vulnerability exploited? • Persistence? • Data accessed?	• Disable Bluetooth where unused • Patch firmware • Disable discoverable mode • Brief users	TP = Bluetooth exploit + access. BP = patched. FP = legitimate pairing.	Disable. Patch. Brief.	Device Bluetooth logs. Pairing history. Vulnerability data.
129	**RFID / Badge Cloning**	**Medium**	Attacker clones employee badge to gain physical access.	• Badge use at unusual hours/locations? • Same badge ID at two locations near-simultaneously? • User reports badge lost? • Camera footage shows different person?	• Badge tech (low-frequency, vulnerable to cloning)? • Other badges similarly cloned? • Physical security perimeter breach? • Insider involvement?	• Disable cloned badge • Re-issue with stronger tech (HID iCLASS SE, MIFARE DESFire) • Coordinate physical security • Tailgating awareness	TP = clone + unauthorised access. BP = caught at access. FP = user shared badge (policy issue).	Disable. Reissue. Phys-sec coordinate.	Badge access logs. Camera footage. Badge tech audit.
130	**Hardware Keylogger / USB Implant**	**High**	Physical device installed on keyboard/USB to capture keystrokes or inject commands.	• Unfamiliar USB device between keyboard and PC? • Anomalous keyboard input pattern? • User reports unexpected input?	• Device type (keylogger, BadUSB, RubberDucky)? • Data captured? • Physical access trail? • Other workstations affected?	• Remove device • Reset creds entered • Image affected workstation • Phys-sec investigation	TP = device + capture. BP = caught at hardware audit. FP = legitimate dongle.	Remove. Reset. Image. Phys-sec.	Physical device. EDR logs (USB events). Camera footage. User keystroke pattern.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• Camera footage of unauthorised access?					
131	**Tailgating / Unauthorised Physical Entry**	**Medium**	Attacker follows authorised person into building or controlled area.	• Security or user reported tailgating? • Camera footage shows unbadged entry? • Access logs show fewer entries than people seen? • Sensitive area breach?	• Attacker captured on camera? • Access to sensitive systems / docs? • Insider assistance? • Data or hardware taken?	• Phys-sec investigation • Access policy reinforcement • Mantraps for high-security areas • Awareness training	TP = tailgate + unauthorised activity. BP = caught at entry. FP = legitimate visitor not properly badged.	Phys-sec. Investigate. Reinforce.	Access logs. Camera footage. User reports. Physical evidence.

Cryptography & PKI (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
132	**Weak / Deprecated Cryptography Use**	**Medium**	App or service using broken algorithms (MD5, SHA1, RC4, DES, SSLv3).	• Vuln scanner flagging weak crypto? • TLS handshake using deprecated cipher? • Cert with SHA1 signature? • Compliance report flagging?	• Where is the weak crypto in use? • Risk to data (hashes for password, MAC, transit)? • Migration plan exists? • Legacy clients depending on it?	• Migrate to strong crypto (AES-256, SHA-256+, TLS 1.2+) • Disable weak ciphers • Update certs • Compliance remediation	TP = weak crypto + risk. BP = config error caught. FP = isolated test environment.	Disable weak. Migrate. Update certs.	Vuln scan. Config audit. Cipher inventory.
133	**Certificate Theft / Misuse**	**High**	Code-signing or TLS cert stolen and used to sign malware or impersonate services.	• Cert reported compromised? • Anomalous signing events? • Cert used to sign unknown binaries? • Cert appearing in malware analysis?	• Source of theft (CA, code-signing system, dev workstation)? • Binaries signed with cert (malware list)? • Persistence using signed binaries? • Revocation status?	• Revoke cert • Issue new cert with hardware backing (HSM) • Hunt for malware signed with cert • Coordinate with CA	TP = cert theft + abuse. BP = caught and revoked. FP = legitimate signing.	Revoke. New cert HSM. Hunt malware.	Cert issuance and signing logs. CA records. Binaries signed. Threat intel reports.
134	**Rogue Internal CA / Sub-CA**	**Critical**	Attacker installs or registers rogue CA to issue trusted certs for MitM or impersonation.	• New CA cert in trusted store? • Cert chain anomaly in TLS interception? • Internal CA-issued certs for external domains? • Endpoint trust store changes?	• Source of CA install? • Certs already issued by rogue? • Persistence via cert? • TLS interception in progress?	• Remove rogue CA from all trust stores • Revoke any issued certs • Reset connections that may have been intercepted • ADCS / PKI hardening	TP = rogue CA + issuance. BP = caught at install. FP = legitimate enterprise CA.	Remove from trust. Revoke certs. Reset connections.	Trust-store changes. CA issuance logs. TLS connection logs. Endpoint EDR.
135	**Padding Oracle / Cryptographic Implementation Flaw**	**Medium**	Attacker exploits flaw in crypto implementation (BEAST, CRIME, padding oracle) to decrypt data.	• Vuln scanner flagging known crypto vuln? • TLS / cipher implementation outdated? • Many small variations of similar request? • Application using non-standard crypto?	• Successfully exploited (data decrypted)? • Volume of probing requests? • Patch available and missing? • Sensitive data at risk?	• Patch / disable affected ciphers • Migrate to authenticated encryption (AEAD) • Audit encrypted data • Tighten WAF / app rules	TP = flaw + exploitation. BP = caught early. FP = legitimate crypto operations.	Patch. Disable. Migrate.	Web logs. App logs. Vuln scan. Cipher audit.
136	**Key Material Theft (Private Keys)**	**Critical**	TLS private keys, SSH keys, or signing keys stolen from server or vault.	• File access alert on key store? • Vault audit showing key export? • Anomalous server access? • Keys appearing in dark-web monitoring?	• Which keys (TLS, SSH, signing)? • Where stolen from (vault, file, memory)? • Used for MitM or signing yet? • Persistence created with key?	• Rotate all affected keys • Revoke certs based on stolen keys • Hunt for misuse • Move to HSM-backed keys	TP = key exfiltration + use. BP = exfil attempt blocked. FP = legitimate key access.	Rotate. Revoke. HSM.	File/vault access logs. Server EDR. Cert/key issuance and use. Network logs.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
137	**Quantum 'Harvest Now, Decrypt Later'**	**Low**	Adversary captures encrypted traffic now, intends to decrypt with future quantum computing.	• High-value encrypted traffic captured by suspicious source? • Long-term sensitive data using non-quantum-resistant algorithms? • Specific targeted captures by nation-state pattern? • Compliance flagging crypto-agility gaps?	• What data has long-term sensitivity (years)? • Current crypto algorithms in use? • Migration plan for post-quantum? • Inventory complete?	• Begin crypto-agility planning • Inventory cryptographic assets • Migrate long-term-sensitive data to PQC algorithms (Kyber, Dilithium) • Vendor coordination	TP = capture confirmed + long-term sensitivity. BP = mitigated by re-encryption. FP = no real concern (short-lived data).	Re-encrypt with PQC. Inventory. Plan migration.	Network captures. Crypto inventory. Vendor PQC roadmaps.

Container & DevOps (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
138	**Malicious Container Image**	**High**	Attacker pushes malicious image to registry, or developer pulls compromised public image.	• Image scan failing in CI? • Image from untrusted registry? • Outbound from container at runtime to suspicious destination? • Backdoor in image layer?	• Image content (cryptominer, RAT, exfil)? • Persistence in image used in production? • Other deployments using same image? • Source of pull (typosquat, registry compromise)?	• Remove image from registry and clusters • Reimage affected nodes • Roll forward to clean image • Tighten registry policy (signed images, allow-list)	TP = malicious image + deployment. BP = caught at scan. FP = legit image with security tool detection.	Remove. Reimage. Sign images. Allow-list.	Image layers. Registry logs. Deployment history. Runtime EDR.
139	**Kubernetes Misconfiguration Abuse**	**High**	Attacker exploits exposed K8s API, anonymous access, or weak RBAC.	• K8s API accessible from internet? • Anonymous access enabled? • Anomalous kubectl from external IP? • Privileged pod creation?	• Specific misconfig (no auth, weak RBAC, hostPath, privileged)? • Cluster compromise level? • Resource hijacking (mining)? • Persistence (DaemonSet)?	• Lock down API (private endpoint, RBAC) • Audit and tighten RBAC • Pod Security Standards enforced • Hunt persistence	TP = misconfig + abuse. BP = caught at audit. FP = misconfigured but not exploited.	Lock down. RBAC. PSS.	K8s audit logs. RBAC config history. Workload inventory. Network policy.
140	**Secrets in Source Code / Pipeline**	**High**	API keys, passwords, tokens committed to repo or pipeline logs.	• Secret scanner alert (GitGuardian, TruffleHog)? • High-entropy strings in commits? • Public repo exposure? • Token use from unexpected source post-leak?	• Which secrets exposed and scope? • When committed (history search)? • Used by attacker yet? • Other repos with same pattern?	• Rotate all exposed secrets • Force history rewrite or repo deletion • Audit usage of exposed secrets • Implement pre-commit hooks	TP = secret leaked + abuse. BP = caught at scan. FP = test/dummy secret.	Rotate. Rewrite history. Pre-commit hooks.	Repo history. Secret-scanner logs. Cloud audit (key use). Public exposure timeline.
141	**Infrastructure-as-Code (IaC) Tampering**	**High**	Attacker modifies Terraform/CloudFormation/Pulumi to add backdoors or weaken security.	• Anomalous IaC commits? • Reduced security controls in deployed infra? • New unauthorised resources? • Pipeline applying	• Source of commit (compromised dev account)? • Deployed infra compromised? • Persistence? • Spread to other repos?	• Roll back IaC • Audit deployed resources • Reset pipeline secrets • Strengthen review gates	TP = malicious IaC + deployment. BP = caught at PR review. FP = legitimate refactor.	Rollback. Reset. Review gates.	Repo history. Pipeline logs. Cloud audit. Resource inventory.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				unreviewed code?					
142	**Build System Compromise (Jenkins/GitHub Actions/Azure DevOps)**	**Critical**	Build server compromised, allowing arbitrary code execution and release tampering.	• Build agent showing malware? • Anomalous job execution? • New unauthorised pipeline secrets? • Outbound from build to suspicious destination?	• Initial vector (exposed Jenkins, plugin vuln, token leak)? • Releases tampered? • Persistence in agents? • Multiple build systems affected?	• Halt builds. Image build agents • Rotate all pipeline secrets • Audit recent releases • Coordinate downstream	TP = build compromise + release tampering. BP = caught early. FP = misconfig.	Halt. Image. Rotate. Audit.	Build logs. Agent EDR. Pipeline history. Release artifacts.
143	**Service Account Token Abuse (K8s/Cloud)**	**High**	Attacker steals service account JWT or token from pod/instance metadata for cluster/cloud access.	• Service account auth from outside cluster? • IMDS access from compromised pod? • Anomalous API calls by service account? • Token exfil from pod logs?	• Token scope and reach? • Persistence created via SA? • Lateral movement to cloud? • IRSA / Workload Identity misconfig?	• Rotate SA tokens • Tighten IRSA / Workload Identity • Egress restrictions on pods • Hunt persistence	TP = SA token abuse externally. BP = blocked by network policy. FP = legit cross-cluster.	Rotate. Tighten identity. Egress restrict.	K8s audit. Cloud audit. Pod logs. Network policy.

AI & Emerging (7)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
144	**AI-Generated Phishing**	**High**	Phishing emails or messages generated by LLM — perfect grammar, contextually personalised, hard to spot.	• Email passes basic gates but content suspicious to user? • Sender + content unusual mismatch? • Lure references real internal context (LinkedIn-scraped)? • User reports message that 'feels off'?	• AI-generation indicators (overly polished, generic exec name)? • Coordinated multi-channel (email + LinkedIn DM)? • Targeting pattern across org? • Detection by content classifier?	• Block sender • Update awareness training (visual cue, AI-aware tactics) • Tune detection for AI-generated content patterns • Brief targets	TP = AI-phish + intent. BP = blocked. FP = legitimate marketing.	Block. Brief. Update training.	Email + headers. Content classifier output. Multi-channel campaign reconstruction.
145	**Deepfake Voice / Video (Vishing 2.0)**	**Critical**	AI-cloned voice or video impersonating exec for fraud (BEC, wire transfer).	• Call/video from 'executive' requesting money or sensitive action? • Voice slightly off (cadence, tone)? • Incoming from unusual number/location? • User reports it didn't sound right?	• Audio analysis for synthesis indicators? • Coordination with legitimate exec? • Wire/payment system halt possible? • Other targeted users?	• Halt any triggered transactions • Verify with exec via independent channel • Brief on deepfake awareness • Implement out-of-band verification for finance	TP = deepfake + action attempted. BP = caught and verified. FP = legitimate exec call.	Halt transactions. Out-of-band verify. Brief.	Audio/video sample. Caller attribution. Verification trail. Finance system logs.
146	**Prompt Injection (LLM Application Abuse)**	**High**	Attacker injects instructions into LLM input to make it ignore guardrails or leak data.	• LLM app behaviour anomalous (refusing tasks, leaking data)? • User input contains instructions ('ignore previous', 'pretend')? • Document or webpage content reaching LLM with malicious instructions (indirect injection)? • Output contains data it shouldn't?	• Direct vs indirect injection? • Data leaked or actions triggered? • System prompt exposure? • Multiple users affected?	• Patch (input filtering, system-prompt isolation, output validation) • Audit interactions • Tighten LLM permissions / tool calling • Update guardrails	TP = injection + leak/action. BP = blocked by guardrail. FP = legit unusual prompt.	Patch. Tighten guardrails. Audit.	LLM interaction logs. Input + output samples. Tool-calling history. System prompts.
147	**Data Poisoning (ML Model Attack)**	**Medium**	Attacker poisons training data to make ML model misbehave (backdoor, bias, evasion).	• Model performance regression? • Training data integrity check failures? • Specific inputs causing odd outputs (backdoor signature)? • Source of training data compromise?	• Type (poisoning, backdoor, evasion)? • Production impact? • Source of poisoned data? • Other models affected?	• Retrain with verified data • Audit data pipeline • Implement data integrity checks • Adversarial robustness testing	TP = poisoning + impact. BP = caught at validation. FP = data quality issue.	Retrain. Audit pipeline. Robustness testing.	Training data lineage. Model version history. Output anomaly logs.
148	**Malicious AI Tool Abuse**	**Medium**	Attackers using purpose-built malicious LLMs to	• Scale and quality of phishing campaigns suddenly higher?	• Campaign scale and targets?	• Adapt detection to AI-scale and sophistication	TP = AI-tool campaign confirmed by TI.	Update detections. Awareness. TI.	Campaign artefacts. TI reports.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
	(FraudGPT, WormGPT)		scale phishing, malware, BEC.	• Malware code with no clear human author signatures? • BEC and social-engineering content sophistication up? • TI reports mentioning specific malicious-AI services?	• Patterns suggesting AI generation? • Customer/partner impacted? • TTP shifts from baseline?	• User awareness emphasising new tactics • TI integration to follow malicious AI services • Brief leadership	BP = absorbed by current defences. FP = standard cybercrime.		Content analysis.
149	**AI Agent Hijacking (Agentic AI Abuse)**	**High**	Attacker hijacks an autonomous AI agent (e.g. workflow agent) to perform unauthorised actions.	• Agent performing actions outside scope? • Agent tool calls to unexpected destinations? • Spike in agent activity? • User reports agent doing things they didn't ask?	• Hijack vector (prompt injection, compromised credential, supply chain)? • Actions taken (data exfil, transactions)? • Agent credentials compromised? • Other agents affected?	• Halt agent • Revoke agent credentials • Audit actions • Tighten agent guardrails and permissions	TP = hijack + unauthorised action. BP = caught early. FP = misconfigured but legitimate.	Halt. Revoke. Tighten.	Agent action logs. Tool call history. Credential audit. Prompt history.
150	**Adversarial ML / Model Evasion**	**Medium**	Attacker crafts input designed to fool ML-based detection (malware classifiers, anti-fraud).	• Detection bypassed by samples that should be flagged? • Adversarial perturbation patterns in samples? • Model accuracy dropping on production data? • Red-team or pen-test reports?	• Evasion technique? • Production miss rate? • Robustness testing results? • Model retraining needed?	• Retrain with adversarial examples • Add ensemble/secondary detection • Robustness testing in CI • Vendor coordination if commercial model	TP = evasion + miss in production. BP = caught by ensemble. FP = legitimate sample.	Retrain. Ensemble. Robustness.	Sample analysis. Model logs. Detection metrics. Robustness test results.

Email Security (Advanced) (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
151	**Email Header Injection / SMTP Smuggling**	**Medium**	Attacker abuses SMTP parsing inconsistencies to smuggle additional emails or bypass auth.	• Mail gateway alerting on header anomalies? • Multiple From or Subject headers in one message? • CRLF injection patterns in mail body? • DMARC/DKIM passing for one but not all?	• Smuggled emails delivered separately? • Recipients targeted with smuggled content? • SMTP servers in chain inconsistent? • Vendor patch available?	• Patch SMTP gateway • Update mail flow rules to drop ambiguous messages • Hunt for previously delivered smuggled emails • Tighten DMARC	TP = smuggling confirmed + bad payload. BP = blocked. FP = malformed legit mail.	Patch. Quarantine messages. DMARC tighten.	Mail gateway logs. Raw message captures. Header analysis. Recipient tracking.
152	**ATP / Sandbox Evasion in Email**	**High**	Malicious attachment crafted to evade sandbox detonation (delay, environment-aware).	• Document or installer that's clean in sandbox but flagged later? • Time-delayed payload activation? • Geofencing or anti-VM checks in sample? • User-interaction required to trigger?	• Evasion techniques used? • Payload after evasion? • Other recipients of same attachment? • Sandbox needs tuning?	• Re-detonate with extended sandbox + interaction simulation • Block payload IOCs • Hunt across estate • Tune sandbox	TP = evasion + post-delivery activation. BP = caught at re-detonation. FP = legit complex installer.	Block IOCs. Hunt. Tune sandbox.	Attachment sample. Sandbox runs (multiple). EDR on opener. Payload analysis.
153	**DKIM/SPF/DMARC Bypass Techniques**	**Medium**	Attacker crafts mail to pass auth despite spoofing (subdomain abuse, DKIM replay).	• DMARC report showing pass on suspicious mail? • From and DKIM-signing domains different? • Reply-to or display-name social engineering? • Subdomain authenticated but not main brand?	• Technique (DKIM replay, subdomain takeover, hijacked partner)? • Volume of abuse? • Targeted recipients? • Coordinate with auth providers?	• Tighten DMARC (subdomain policy) • Investigate compromised signing keys • Block specific abuse patterns • Auth provider coordination	TP = bypass + bad payload. BP = caught by content scan. FP = legit subsidiary.	DMARC tighten. Block. Investigate.	Mail headers. DMARC reports. DNS records. Content analysis.
154	**Conversation Hijacking (Thread-Jacking)**	**High**	Attacker injects malicious reply into ongoing legitimate conversation.	• Reply to existing thread but sender slightly off? • Lookalike domain or display-name swap? • Unexpected attachment in mid-thread reply? • Tone or request shift in latest reply?	• Original participant compromised? • Vendor mailbox compromise (their side)? • Other thread members targeted? • Pattern across multiple threads (mass harvest)?	• Notify all thread participants • Block sender + payload • Vendor security coordination • Hunt for hijacked threads across org	TP = hijack + bad payload. BP = caught. FP = legit thread continuation.	Block. Notify. Vendor coord.	Thread + hijacked reply. Mailbox audit. Vendor coordination notes. IOC hunt.
155	**Mailbox Auto-Forward Exfiltration**	**High**	Compromised mailbox set to auto-forward all incoming	• External auto-forward rule on mailbox?	• Volume of exfil over time? • Sensitive content forwarded?	• Disable rule + reset password + revoke sessions	TP = unauthorised forward + exfil. BP = caught at	Disable rule. Reset. Revoke. Tenant policy.	Mailbox audit (rule history). Sign-in logs. Forwarded

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
			mail externally for ongoing recon.	• Created from unusual session/location? • Sustained outbound forwarding traffic? • User unaware of rule?	• Other mailboxes with same pattern? • Tied to BEC or recon campaign?	• Block external forwarding tenant-wide • Hunt all mailboxes for similar • Audit forwarded content	rule creation. FP = legit user-set forward.		content. Source IPs.
156	**Calendar / Meeting Invite Phishing**	**Medium**	Phishing via calendar invite (auto-accepted in some clients) with malicious link or join URL.	• Calendar invite from unknown sender? • Join URL or attachment in invite suspicious? • Auto-accept on user's calendar? • Multiple users got similar invite?	• Did user click join URL? • Cred harvest or malware delivery? • Source spoofing or compromised account? • Mass campaign?	• Block sender + URL • Disable calendar auto-accept by policy • Brief users • Hunt for similar across estate	TP = bad invite + click. BP = invite blocked. FP = legit external meeting.	Block. Disable auto-accept. Brief.	Calendar audit. Invite content. Click logs. Source attribution.

Database Attacks (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
157	**NoSQL Injection (MongoDB, CouchDB, etc)**	**High**	Attacker injects operators ($ne, $gt, $where) into NoSQL queries to bypass auth or extract data.	• WAF alert for NoSQL operator patterns? • Anomalous responses to login or search? • Unexpected JSON in input fields? • DB driver errors revealing structure?	• Injection target (auth bypass, data extraction)? • Successful exploitation? • Volume of malicious requests? • DB type and version?	• Patch app (parameterised queries, schema validation) • Audit data accessed • Tighten WAF • Reset compromised accounts	TP = NoSQL inj + data access. BP = blocked. FP = legit JSON input.	Patch. WAF. Reset.	WAF + web logs. App logs. DB query logs (if available). Source IP.
158	**Database Privilege Escalation**	**High**	Attacker exploits DB account or stored procedure to gain DBA or sysadmin privilege.	• Privilege change in DB audit log? • sp_addsrvrolemember or GRANT events from low-priv account? • xp_cmdshell enabled or used? • Vulnerable stored proc invoked?	• Initial access (SQLi, weak account)? • Privilege gained and used for what? • OS-level command exec via xp_cmdshell? • Persistence?	• Revoke escalated privilege • Disable xp_cmdshell • Patch DB and apps • Reset DB accounts	TP = priv esc + abuse. BP = blocked. FP = sanctioned admin work.	Revoke. Disable xp_cmdshell. Patch. Reset.	DB audit. Stored proc audit. Login history. Server EDR.
159	**Database Backup / Snapshot Theft**	**Critical**	Attacker exfiltrates DB backup file (often plaintext copy of all data).	• Large file transfer from DB or backup server? • Backup file accessed by non-DBA? • Cloud backup downloaded externally? • Backup encryption weak or missing?	• What data in backup? • How was it accessed (cred theft, share misconfig)? • External destination? • Backup chain integrity?	• Rotate any creds in backup • Notify affected parties / regulators • Audit backup access • Encrypt all backups + access tighten	TP = backup exfil. BP = caught at egress. FP = legit DBA copy.	Rotate creds. Audit. Encrypt + access tighten.	File access logs. Backup chain. Cloud audit. Data inventory.
160	**DB Connection String Leak**	**High**	Connection string with creds exposed in code, config file, log, or error message.	• Secret-scanner alert? • Config file in public repo? • Stack trace in app log exposing creds? • Connection from unusual IP using leaked creds?	• Credentials still valid? • Used by attacker yet? • Other secrets co-located? • Access scope?	• Rotate DB creds • Audit DB access logs • Move secrets to vault • Pre-commit hooks	TP = leak + use. BP = found before use. FP = test creds.	Rotate. Vault. Audit.	Secret-scan logs. Repo history. DB access logs. Vault audit.
161	**Mass Data Extraction via App**	**High**	Attacker uses app legitimately but at abuse scale to scrape DB.	• App API hit at unusual volume by one identity? • Sequential ID enumeration? • Result-size patterns suggesting bulk extract?	• Account compromise vs malicious authorised user? • Volume and content extracted? • Rate-limit gaps? • Other identities doing similar?	• Rate limit + abuse detection • Tighten API access patterns • Investigate identity • Audit data extracted	TP = abuse + data extracted. BP = rate limit. FP = legit bulk export.	Rate limit. Investigate. Audit.	API logs. Auth logs. Result size patterns. Identity audit.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• Authenticated but abusing scope?					
162	**DB Tampering / Data Integrity Attack**	**Critical**	Attacker modifies DB records to commit fraud, sabotage, or cover tracks.	• UPDATE/DELETE events outside business hours? • DBA-level modifications without ticket? • Specific high-value records changed? • Audit log gaps or tampering?	• What records changed? • Who and from where? • Audit table modified? • Persistence or one-off?	• Restore from clean backup • Audit all changes via diff with backup • HR/Legal coordination if insider • Tighten DB audit + alerting	TP = unauthorised DB change. BP = caught at trigger. FP = sanctioned change.	Restore. HR/Legal. Tighten audit.	DB audit logs. Backup comparison. Identity history. Audit log integrity check.

API Attacks (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
163	**Broken Object Level Authorization (BOLA / IDOR)**	**High**	API allows access to objects (orders, profiles, files) by ID without checking ownership.	• API requests to /api/users/{id} with sequential IDs? • One user's session accessing another's resources? • No 403 returned for foreign-object access? • Bug bounty or pen-test finding?	• Scope of exposure (PII, financial, health)? • Successful access count? • Pattern of enumeration? • Attribution?	• Patch (object-level access checks server-side) • Audit data accessed • Notify affected users • Tighten testing	TP = BOLA + access. BP = blocked. FP = legit shared resource.	Patch. Audit. Notify.	API logs. Auth + object IDs. Access patterns.
164	**Mass Assignment / Excessive Data Exposure**	**Medium**	API binds client input to model fields blindly, allowing setting protected fields (isAdmin=true).	• JSON requests with extra fields client shouldn't set? • User updates with isAdmin/role/permissions in payload? • Application granting privilege without admin route? • Anomalous role assignments?	• Successful mass assignment? • Privileged accounts created? • Persistence? • Other endpoints similarly vulnerable?	• Patch (allow-list bindable fields) • Audit roles and privileges • Reset compromised accounts • Tighten code review	TP = mass assign + privilege change. BP = blocked. FP = legit field update.	Patch. Audit. Reset.	API logs. Object change history. Auth events.
165	**GraphQL Abuse (Deep Nesting / Introspection)**	**Medium**	Attacker exploits GraphQL features (nested queries, introspection, batching) for DoS or info disclosure.	• Deeply nested GraphQL queries (DoS)? • Introspection enabled in production? • Schema enumeration patterns? • Resource-exhausting queries?	• Performance impact? • Schema disclosed? • Specific attack (introspection + BOLA chain)? • Volume?	• Disable introspection in prod • Query depth + cost limits • Patch resolvers • Rate limiting	TP = abuse + impact. BP = limits caught. FP = legit complex query.	Disable introspection. Limits. Patch.	GraphQL logs. Query analysis. Performance metrics.
166	**API Key / Token Theft**	**High**	Long-lived API key stolen and used.	• Key found in public repo / leak monitoring? • Key used from unusual IP? • Anomalous API call patterns? • Key scope and age?	• What was accessed? • Persistence (new keys, IAM changes)? • Other keys with similar exposure? • Source of leak?	• Rotate key • Audit usage • Move to short-lived tokens / OAuth • Pre-commit hooks + secret scan	TP = leak + abuse. BP = caught at rotation. FP = legit automation.	Rotate. Audit. Short-lived migrate.	API logs. Leak source. Cloud/SaaS audit. Repo history.
167	**Webhook Abuse / SSRF via Webhooks**	**Medium**	Attacker abuses outbound webhook feature (or registers their own) for SSRF, exfil, or amplification.	• Webhook configured to internal IP / metadata service? • Outbound from app tier to unexpected URL? • User registering webhook to private destination? • Volume of webhook fires unusual?	• Internal access via webhook? • Cloud metadata or internal services hit? • Persistence? • Mass abuse for amplification?	• Webhook destination allow-list (no internal/metadata) • Egress restriction from webhook tier • Audit webhook configs • Rate limit webhook fires	TP = webhook abuse. BP = blocked. FP = legit webhook config.	Allow-list. Egress. Audit.	Webhook config history. Outbound logs. Internal access logs.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
168	**Shadow / Undocumented API Discovery**	**Medium**	Attacker finds and abuses undocumented or deprecated endpoints (often without auth or modern controls).	• Requests to endpoints not in current API docs? • Old API versions still active? • Auth bypassed on legacy endpoint? • Bug bounty or recon flagging?	• Functionality of shadow API? • Successful abuse? • Other endpoints similarly exposed? • Inventory complete?	• Inventory all API endpoints • Decommission shadow/deprecated endpoints • Apply consistent security controls • API gateway enforcement	TP = shadow API + abuse. BP = inventoried + secured. FP = legit internal endpoint exposed by mistake.	Inventory. Decommission. Gateway.	Web/API logs. Endpoint inventory. Source patterns.

Reconnaissance & OSINT (5)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
169	**External Surface Scanning**	**Low**	Attacker scans your public-facing infrastructure (Shodan-style or active scan).	• High volume of probes from one source? • Scanner signature in logs (Nmap, Masscan, ZMap)? • Targeted ports/services? • Known scanning ASN?	• What's exposed (services, versions)? • Vulnerable version targeted? • Followed by exploitation? • Attribution?	• Reduce attack surface (close unused services) • Patch exposed services • Block known bad scanners • ASM (attack surface management) program	TP = scan + targeted vuln. BP = absorbed. FP = legit research scan.	Reduce surface. Patch. Block.	Firewall + edge logs. Scanner attribution. Service inventory.
170	**Dark Web / Leaked Credential Monitoring**	**Medium**	Employee credentials appearing on dark web from third-party breaches.	• TI feed alert for corp domain emails? • HIBP / SpyCloud match? • Credentials with passwords or hashes? • Recent breach affecting partner or service?	• Active accounts with leaked passwords? • Reuse on corporate? • MFA in place? • Used by attacker yet?	• Force password reset on matching accounts • Enforce MFA • Block password reuse • Continuous monitoring	TP = leak + still-valid corp credential. BP = monitoring caught it. FP = old/test creds.	Reset. MFA. Block reuse.	TI feed records. HIBP/SpyCloud. Auth logs.
171	**Social Media Recon / Spear-Phish Prep**	**Low**	Attacker building targeting profile from LinkedIn, Twitter, conference talks.	• TI report mentioning your org targeted? • Suspicious LinkedIn connection requests to execs? • Fake recruiter outreach? • Pretext-building patterns?	• Specific employees targeted? • Followed by phishing? • Profile fakeness indicators? • Coordinated campaign?	• Brief targeted employees • Take down fake profiles (LinkedIn / Twitter security) • Update awareness training • Enhanced monitoring on targets	TP = recon + follow-up attack. BP = identified early. FP = legit recruiter.	Brief. Take down. Monitor.	Social media records. Communication logs. Subsequent attack correlation.
172	**DNS / Subdomain Enumeration**	**Low**	Attacker enumerates DNS records and subdomains to find forgotten or weak assets.	• DNS query patterns suggesting brute-force enumeration? • Cert transparency log monitoring showing recon? • Subdomains with weak posture being probed? • Tools (Sublist3r, Amass) signatures?	• Forgotten subdomain compromised? • Subdomain takeover attempted? • Vuln subdomain exploited? • Attribution?	• Inventory all subdomains • Decommission unused • Subdomain takeover protection • Continuous monitoring	TP = enum + abuse. BP = inventory + secure. FP = legit research.	Inventory. Decommission. Monitor.	DNS logs. Cert transparency. Subdomain inventory.
173	**GitHub / Public Code Mining**	**Medium**	Attacker mines public GitHub for secrets, internal code, or recon info.	• Secret-scanner or BinaryEdge alert? • Internal code / docs in public repos? • Employees pushing to personal accounts?	• What was leaked? • Active credentials? • Internal architecture exposed? • Pattern across multiple repos?	• Rotate exposed secrets • Engage GitHub Trust & Safety for takedown • Brief employees on policy • Pre-commit hooks org-wide	TP = exposure + abuse risk. BP = found at scan. FP = sanctioned open-source.	Rotate. Takedown. Hooks.	GitHub search. Repo history. Secret-scan logs.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• Targeted searches for your domain in code?					

Living-off-the-Land (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
174	**PowerShell Abuse (Encoded / Downgrade)**	**High**	Attacker uses PowerShell with encoded commands, downgrade to v2, or AMSI bypass.	• powershell.exe -EncodedCommand or -nop -w hidden patterns? • PowerShell v2 invocation (-Version 2)? • AMSI bypass strings in script? • Parent process unusual?	• Decoded command — what does it do? • Downloads / executes payload? • Persistence? • Lateral movement?	• Disable PowerShell v2 • Constrained Language Mode + AMSI • Script-block + module logging • Block known bypass patterns	TP = encoded + bad action. BP = blocked. FP = admin script.	Disable v2. CLM. Block patterns.	PowerShell logs (4103/4104). EDR command lines. Decoded scripts.
175	**WMI Abuse (Wmic / Invoke-WmiMethod)**	**High**	Attacker uses WMI to execute, persist, or query remotely.	• wmic process call create from unusual source? • Invoke-WmiMethod with payload? • WMI subscription event consumer added (persistence)? • Remote WMI to many hosts?	• Source compromised? • Lateral movement pattern? • Persistence via WMI? • Cred reuse?	• Block remote WMI at FW • Hunt WMI subscriptions • Reset creds • Tighten EDR for WMI patterns	TP = WMI abuse. BP = blocked. FP = sanctioned automation.	Block. Remove subs. Reset.	WMI activity logs. Sysmon Event 19/20/21. EDR.
176	**Certutil / Bitsadmin Download**	**Medium**	Attacker uses certutil -urlcache or bitsadmin to download payload from internet.	• certutil with -urlcache or -decode arguments? • bitsadmin /transfer to external URL? • Process spawning these from non-admin context? • File written to suspicious path?	• Downloaded payload — malware? • Decoded base64 from script? • Followed by execution? • Persistence?	• Application control blocking these binaries from non-admin • Block patterns at EDR • Hunt for downloads • Reimage	TP = certutil/bitsadmin + download + execute. BP = blocked. FP = admin tool use.	Block. Hunt. Reimage.	EDR cmdline. Network connections. Files written.
177	**Mshta / Rundll32 / Regsvr32 Abuse**	**High**	Attacker uses Windows utilities to execute scripts or DLLs while bypassing AppLocker.	• mshta with URL or HTA file? • rundll32 with unusual arguments / DLL paths? • regsvr32 /s /u /i scrobj.dll (Squiblydoo)? • Parent process Office or scripting?	• What was loaded/executed? • Squiblydoo / Squiblytwo patterns? • Persistence? • Hunt across estate?	• Application control rules • Hunt for these LOLBin patterns • Tighten EDR • Reimage if executed	TP = LOLBin + bad action. BP = blocked. FP = legit installer.	AppControl. Hunt. EDR.	EDR cmdline. Sysmon Event 1. Network. Files loaded.
178	**MSBuild / InstallUtil / Csc Abuse**	**Medium**	Attacker uses .NET dev tools to compile/execute code in-memory, bypassing AV.	• MSBuild loading XML from unusual location? • InstallUtil /logfile= /U pattern? • csc.exe compiling from temp / suspicious path? • Parent unusual?	• Compiled / loaded code — what does it do? • Persistence? • Cred theft? • Lateral movement?	• Application control block • Hunt for these patterns • EDR tuning • Reimage	TP = .NET LOLBin abuse. BP = blocked. FP = developer activity.	Block. Hunt. Reimage.	EDR. Compiled artefacts. Source XML. Network.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
179	**ScheduledTasks / At Abuse for Execution**	**Medium**	Attacker creates scheduled task as execution or persistence mechanism (often via schtasks /create).	• schtasks /create from unusual source? • Task created with SYSTEM context? • Task action runs script from temp / writable path? • Many hosts getting same task name?	• Task action — payload? • Persistence intent? • Created remotely? • Lateral pattern?	• Remove tasks • Hunt across estate • Reimage if needed • EDR tuning for task creation	TP = task + bad payload. BP = blocked. FP = legit IT scheduled job.	Remove. Hunt. Reimage.	4698 events. Task definitions. EDR cmdline. File system.

Defense Evasion (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
180	**Log Clearing / Tampering**	**High**	Attacker clears Windows event logs or tampers with logging to hide actions.	• Event 1102 (Security log cleared)? • Event 104 (System log cleared)? • wevtutil cl with privilege? • Audit policy change events (4719)?	• Who cleared and when? • Other logs tampered? • Hands-on attacker indicators? • Centralised logging caught the cleared events?	• Restore from central log archive • Treat as IR — assume compromise • Forensic image • Tighten log forwarding	TP = unauthorised clear. BP = blocked by audit policy. FP = sanctioned admin (rare).	Image host. IR. Strengthen log forwarding.	1102/104 events. Central log archive (SIEM). EDR. Audit policy history.
181	**AV / EDR Tampering**	**Critical**	Attacker disables, uninstalls, or blinds AV/EDR to operate freely.	• EDR offline alert? • AV service stopped on host? • Driver tampering signatures (e.g. EDRSilencer)? • BYOVD (bring your own vulnerable driver)?	• Method (kill, unload driver, BYOVD)? • Hands-on attacker? • What did they do post-disable? • Other hosts targeted?	• Re-enable EDR + reimage • Hunt for actions during blind window • Tier-0 lockdown • Vendor coordination	TP = tampering + post-tamper actions. BP = self-protection blocked. FP = sanctioned uninstall.	Reimage. Tier-0 lockdown. Hunt blind window.	EDR last-seen. Driver loads. Memory. Reconstruct blind window from network/SIEM.
182	**Timestomping**	**Medium**	Attacker modifies file timestamps to evade time-based forensic correlation.	• Files with $STANDARD_INFORMATION mismatched to $FILE_NAME timestamps? • Files in system folders with future or back-dated timestamps? • Tools (timestomp, SetMACE) signatures? • Suspicious file with normal-looking dates?	• What files altered? • Tied to known attack timeline? • Anti-forensics goal? • Persistence?	• Use $FILE_NAME timestamps for true creation • Hunt for tampered files • File integrity monitoring • Forensic deeper review	TP = stomped + bad file. BP = caught. FP = software install with weird timestamps.	Forensics. Hunt. FIM.	$MFT analysis. File metadata both attribute sets. EDR file events.
183	**DLL Search Order Hijack / Side-Loading**	**Medium**	Attacker drops DLL where vulnerable app loads it, gaining persistence in trusted process.	• Trusted app loading DLL from unusual / writable path? • Unsigned DLL loaded by signed binary? • Newly written DLL in app directory? • Process spawning anomalously?	• Vulnerable app abused? • Payload? • Persistence? • Same DLL across hosts?	• Patch app load order • Remove malicious DLL • Hunt across fleet • FIM on app dirs	TP = side-load + payload. BP = blocked. FP = legit plugin DLL.	Patch. Remove. Hunt.	DLL signature. EDR DLL load events. App config. File system.
184	**Masquerading (Renamed Binaries)**	**Medium**	Malicious tool renamed to look like legitimate Windows binary (svchost.exe, lsass.exe in wrong path).	• svchost.exe outside System32? • Process name matching common system bin from non-standard path? • Hash mismatch for known system binary? • Anomalous parent for renamed bin?	• Real malware identification? • Persistence? • Hands-on attacker? • Other masqueraded bins?	• Reimage host • Hunt for masquerading patterns • EDR tuning • Block hash	TP = masquerade + bad behaviour. BP = caught. FP = developer tool oddly placed.	Reimage. Hunt. Block.	EDR process metadata (path + hash + signer). Sysmon Event 1.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
185	**Hidden / Alternate Data Streams (ADS)**	**Low**	Attacker stores payload in NTFS Alternate Data Stream — hidden from normal directory listing.	• EDR detection of ADS write? • File:streamname patterns in cmdline? • Tools (ADS-aware scanners) flagging? • Suspicious file with hidden stream?	• Stream content? • Execution from ADS? • Persistence? • Other files with ADS?	• Remove streams • Hunt across estate • FIM with ADS awareness • Reimage if executed	TP = ADS + payload. BP = blocked. FP = legit Zone.Identifier (download zone).	Remove. Hunt. FIM.	NTFS ADS scan. EDR. Sysmon Event 15 (FileStreamCreate).

Persistence (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
186	**Registry Run Keys / Startup Folder**	**Medium**	Attacker adds entry to Run, RunOnce, or Startup folder to launch on logon.	• Sysmon Event 13 for Run/RunOnce key write? • New file in user / common Startup folder? • Auto-runs from suspicious path? • User-context vs SYSTEM persistence?	• Payload executed by entry? • Other auto-run locations also seeded? • Multiple users? • Same persistence on other hosts?	• Remove entry • Reimage if compromised • Hunt across estate • Sysmon + Autoruns baselining	TP = malicious auto-run. BP = blocked. FP = legit installed app.	Remove. Reimage. Hunt.	Registry / file logs. Autoruns export. Payload sample.
187	**Scheduled Task Persistence**	**Medium**	Attacker creates scheduled task that runs payload on schedule or trigger.	• 4698 (task created) from unusual source/user? • Task action runs from temp / writable / unusual path? • SYSTEM-level task created by non-admin? • Task name resembling legit Windows tasks?	• Action — payload? • Trigger (logon, idle, schedule)? • Created remotely? • Hunt for similar tasks fleet-wide?	• Remove tasks • Reimage if needed • Hunt • Tighten task creation auditing	TP = task + bad payload. BP = blocked. FP = sanctioned IT job.	Remove. Reimage. Hunt.	4698/4699/4700 events. Task XML. EDR. File system.
188	**Service Creation / Modification**	**High**	Attacker installs Windows service or modifies existing one to run payload as SYSTEM.	• 7045 (service installed) for new service from non-admin source? • Service binary path in writable / temp directory? • Existing service binPath modified? • Service name resembling legit ones?	• Service binary — payload? • Auto-start configured? • Created remotely (SCM)? • Other hosts with same service?	• Stop and remove service • Reimage if compromised • Hunt fleet-wide • Service creation alerting	TP = malicious service. BP = blocked. FP = legit installer.	Remove. Reimage. Hunt.	7045 events. Service config. EDR. Binary sample.
189	**WMI Event Subscription Persistence**	**High**	Attacker creates WMI permanent event subscription (filter+consumer+binding) to run on event.	• Sysmon Event 19/20/21 for WMI subscription? • __EventFilter, __EventConsumer, __FilterToConsumerBinding created? • Consumer command-line suspicious? • Triggered on logon, schedule, or process event?	• Consumer payload — what executes? • Persistence stealth (survives reboot, no startup folder)? • Other hosts with same? • Source of creation?	• Remove WMI subscriptions • Reimage if compromised • Hunt fleet-wide for WMI persistence • Sysmon module 19/20/21 enabled	TP = WMI sub + payload. BP = blocked. FP = legit management tool.	Remove. Reimage. Hunt.	Sysmon 19/20/21. WMI repository. EDR.
190	**COM Hijacking**	**Medium**	Attacker registers malicious COM object to be loaded by legitimate app.	• Registry write to HKCU\Software\Classes\CLSID with custom InprocServer32? • DLL path in writable user directory?	• Hijacked CLSID — what triggers it? • Payload DLL?	• Remove registry hijack • Reimage if compromised • Hunt fleet-wide	TP = COM hijack + payload. BP = blocked. FP = legit COM registration.	Remove. Reimage. Hunt.	Registry events. EDR DLL load. Payload sample.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• Process loading hijacked CLSID? • Tools (COMHijackToolkit) signatures?	• Persistence stealth? • Other hosts?	• Registry monitoring			
191	**BITS Job Persistence**	**Medium**	Attacker uses Background Intelligent Transfer Service to download/execute payload persistently.	• bitsadmin /create or /addfile from unusual source? • BITS job with notify command? • Long-running BITS job to suspicious URL? • Sysmon BITS events?	• Job target and command? • Payload? • Persistence (re-fires on event)? • Other hosts?	• Cancel BITS jobs • Reimage if compromised • Hunt fleet-wide • BITS monitoring	TP = BITS persistence. BP = blocked. FP = legit Windows Update.	Cancel. Reimage. Hunt.	BITS event log. EDR. Network. Job XML.

Privilege Escalation (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
192	**UAC Bypass**	**High**	Attacker bypasses User Account Control to elevate from medium to high integrity.	• Known UAC bypass technique signatures (fodhelper, eventvwr, sdclt)? • High-integrity process spawned without consent prompt? • Auto-elevate Windows binary launched from suspicious context? • DLL hijack in elevation flow?	• Technique used? • Followed by privilege actions? • Persistence created at higher integrity? • Hunt fleet-wide?	• Patch (latest Windows updates) • UAC at highest setting • Remove local admin where possible • Hunt for patterns	TP = bypass + post-elevation actions. BP = blocked. FP = legit elevation.	Patch. Reduce admin rights.	EDR process tree. Sysmon. Registry. UAC events.
193	**Token Manipulation / Impersonation**	**High**	Attacker steals or duplicates token of higher-privilege process to act as that user.	• EDR alert for SeImpersonatePrivilege abuse? • Process accessing other process tokens? • Tools (Incognito, JuicyPotato, RoguePotato, PrintSpoofer) signatures? • Anomalous SYSTEM-level activity from non-SYSTEM process?	• Source process and target? • Action taken with stolen token? • Persistence via SYSTEM context? • Hunt across estate?	• Patch (Potato variants need patches) • Reduce privileges granting Impersonate • Reimage compromised host • EDR tuning	TP = token manip + abuse. BP = blocked. FP = legit service impersonation.	Patch. Reduce privs. Reimage.	EDR token events. Sysmon. Process tree.
194	**Kernel Exploit / BYOVD**	**Critical**	Attacker exploits kernel vulnerability or loads vulnerable driver to gain SYSTEM/kernel.	• Driver load event for known-vulnerable driver (HEVD, Capcom, RTCore)? • Kernel crash followed by elevated process? • BYOVD signature in EDR? • Anti-EDR driver loaded?	• Vulnerability exploited? • Persistence at kernel level? • EDR/AV disabled post-exploit? • Other hosts targeted?	• Patch + driver block-list • Reimage host • Hunt for BYOVD across fleet • Vendor (Microsoft Vulnerable Driver Block List)	TP = exploit + kernel. BP = blocked by block-list. FP = legit driver install.	Patch. Block-list. Reimage. Hunt.	EDR driver loads. Memory image. Kernel logs.
195	**Sudo / SUID Abuse (Linux)**	**High**	On Linux, attacker abuses sudo misconfig or SUID binaries to escalate to root.	• Sudo command from unusual user? • SUID binary with shell escape (vim, less, find via GTFOBins)? • Audit log shows escalation pattern? • Newly added SUID files?	• Misconfig (NOPASSWD, ALL)? • Path of escalation? • Persistence as root? • Other hosts similarly misconfigured?	• Fix sudoers config • Remove unnecessary SUIDs • Hunt fleet-wide • auditd tuning	TP = sudo / SUID abuse. BP = blocked. FP = legit admin work.	Fix config. Remove SUIDs. Hunt.	auditd. Sudo logs. File system. SSH/auth logs.
196	**Service Path / Registry Permission Abuse**	**Medium**	Service binary path is unquoted (PathInter) or service config is	• Service with unquoted path containing spaces? • Service registry key writable by non-admin?	• Hijack written? • Service restart triggered for execution?	• Fix service quoting • Tighten service registry permissions	TP = misconfig + exploit. BP = audited and fixed. FP = pen-test finding only.	Fix. Audit. Hunt.	Service config. Registry permissions. EDR. Sysmon.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
			writable, attacker hijacks it.	• Service binary in user-writable directory? • Tools (PowerUp, accesschk) recon traces?	• Other vulnerable services? • Persistence?	• Remove user-writable service paths • Audit fleet			
197	**Stored Credentials / Cleartext on Disk**	**Medium**	Passwords, tokens, or keys found in scripts, INI files, or unattend.xml — used to escalate.	• String search alerts (passwords, tokens) in file system? • unattend.xml, sysprep.inf with creds? • Custom scripts with hardcoded creds? • Group policy preference files (cpassword)?	• Creds still valid? • Reuse across systems? • Persistence using found creds? • Hunt fleet-wide for similar files?	• Reset all found creds • Remove cleartext from scripts • Move to vault / managed identity • Hunt fleet	TP = creds + use. BP = found in audit. FP = test creds.	Reset. Vault. Hunt.	File system. Audit logs of access. Cred history.

Browser Attacks (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
198	**Malicious Browser Extension**	**High**	User installs (or attacker pushes) extension that steals data, injects content, or proxies traffic.	• New extension install on managed browser? • Extension permissions excessive (read/change all data)? • Extension from non-store source (sideloaded)? • User reports browser oddities?	• Extension behaviour (cred theft, ad injection, MitM)? • Data accessed? • Other users with same extension? • Source (compromised dev account, fake)?	• Remove extension via group policy • Reset creds entered while extension active • Brief users • Tighten extension allow-list	TP = malicious + data theft. BP = blocked by policy. FP = legit business extension.	Remove. Reset. Allow-list.	Browser logs. Extension manifest. Network from browser. User cred history.
199	**Browser-in-the-Browser (BitB) Attack**	**Medium**	Phishing page renders fake browser popup mimicking OAuth/SSO login window.	• User reports SSO popup that didn't behave normally? • Page rendering window-like UI inside main page? • URL bar in fake popup not real? • Triggered from phishing or compromised site?	• Creds entered to fake popup? • Domain of phishing site? • Other users targeted? • Linked to AitM phishing?	• Block phishing URL • Reset creds entered • Brief users on BitB indicators • Phishing-resistant MFA (FIDO2)	TP = BitB + creds. BP = caught. FP = legit OAuth popup.	Block. Reset. FIDO2.	Browser history. Phishing site analysis. Sign-in logs.
200	**Drive-By Download / Exploit Kit**	**High**	User visits compromised or malicious site, exploit triggers without user click.	• EDR alert for browser exploit? • Browser process spawning anomalous child? • Connection to known exploit-kit domain? • Patch level on browser/plugins?	• Vulnerability exploited? • Payload delivered? • User aware? • Other users hit same site?	• Reimage host • Patch browser/plugins • Block exploit kit domains • Brief users	TP = exploit kit + payload. BP = blocked. FP = false positive on browser activity.	Reimage. Patch. Block.	EDR timeline. Browser logs. Network. Patch level.
201	**Browser Cookie / Token Theft (Infostealer Adjacent)**	**High**	Malware (or rogue extension) reads browser cookie database for session tokens.	• EDR alert for cookie DB access by non-browser process? • Outbound exfil after cookie read? • Tokens replayed externally? • Infostealer family detected?	• Which sessions stolen? • Cred replay from foreign IP? • Lateral via session? • Browser hardening config?	• Revoke all sessions • Reset creds • Reimage • Move to phishing-resistant MFA	TP = cookie theft + replay. BP = blocked. FP = legit backup tool.	Revoke. Reset. Reimage.	EDR file access. Network exfil. Sign-in logs (replay).
202	**Click-Jacking / UI Redress**	**Low**	Attacker overlays transparent iframe so clicks land on attacker target.	• User reports unintended action? • Iframe of your site embedded on attacker site? • Missing X-Frame-Options / CSP?	• Sensitive action exploitable (transfer, delete)? • Affected users? • Pattern over time? • Mitigation already in place?	• Add X-Frame-Options DENY / CSP frame-ancestors • Audit affected actions • Notify users • Tighten WAF	TP = framing + action triggered. BP = blocked. FP = legit embedding.	X-Frame-Options. Audit. Notify.	Web logs. Referrer. User reports.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• Pen-test finding?					
203	**Tabnabbing / Reverse Tabnabbing**	**Low**	Linked page rewrites the original tab to phishing while user is on the new tab.	• Reports of original tab unexpectedly showing login again? • Pages opened with target=_blank without rel=noopener? • Audit shows vulnerable links? • Pen-test finding?	• Affected users? • Successful credential capture? • Source page (compromised partner)? • Mitigation in place?	• Add rel=noopener noreferrer to all external links • Audit codebase • Brief users • WAF / browser hardening	TP = tabnabbing + creds. BP = blocked by rel=noopener. FP = legit redirect.	Patch. Audit. Brief.	Web logs. Browser history. User reports.

Social Engineering (5)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
204	**Help Desk Social Engineering**	**High**	Attacker calls IT help desk impersonating user to reset password / MFA.	• Help desk reports user 'forgot' MFA? • User reports their MFA changed without them doing it? • Reset triggered from non-standard channel? • Pretext details suspicious (urgency, exec name-drop)?	• What was reset? • Account compromise indicators? • Verification process bypassed? • Tied to broader campaign (Scattered Spider-style)?	• Reverse changes + reset legitimate user • Tighten help desk verification (callback, manager confirm, in-person) • Brief help desk staff • Detect repeats	TP = social engineering + reset + abuse. BP = blocked at verification. FP = legit user.	Reverse. Reset. Tighten verification.	Help desk ticket. Call recording (if any). Account changes. Subsequent activity.
205	**Vishing for IT Access (Scattered Spider-style)**	**Critical**	Attacker calls IT impersonating exec / employee to social-engineer privileged access (often follow-up to recon).	• Call to IT requesting urgent admin access? • Exec impersonation pattern? • Coordinated with phishing or recon? • TI report mentioning the TTPs (Scattered Spider, MGM-style)?	• Access granted? • Hands-on attacker indicators? • Specific high-value targets (cloud admin, AD admin)? • Account compromise scope?	• Revoke any access granted • Activate IR — assume hands-on attacker • Reset all admin accounts • TI sharing + LE engage	TP = SE + access + adversary. BP = blocked. FP = legit IT request.	Revoke. IR. Reset. LE.	Call records. Help desk tickets. Account changes. EDR/SIEM full timeline.
206	**Deepfake Audio Impersonation**	**Critical**	AI-generated voice clone of exec used in phone call for fraud.	• Phone call from 'exec' requesting urgent action? • Voice slightly off or audio quality unusual? • Number spoofed or unfamiliar? • User reports doubt?	• Action requested (wire, data, access)? • Verification via independent channel? • Coordinated with email/SMS? • Other targets?	• Halt any triggered actions • Out-of-band verification with exec • Brief team • Implement callback policy for finance/IT	TP = deepfake + action. BP = caught. FP = legit exec call.	Halt. Verify. Callback policy.	Call records / audio. Exec corroboration. Action audit (wire, IT change).
207	**In-Person / Pretexting Attack**	**Medium**	Attacker physically enters facility posing as vendor, employee, or visitor.	• Camera footage of unbadged or pretexted entry? • Reports of unfamiliar 'IT' / 'auditor' onsite? • Tailgating + pretext combo? • Sensitive access attempted?	• Action taken onsite (USB drop, photos, theft)? • Specific target (server room, exec floor)? • Coordination with external attack? • Other facilities?	• Phys-sec investigation • Brief security guards on pretexting • Visitor management tightened • Vendor verification process	TP = pretexting + action. BP = caught at entry. FP = legit visitor.	Phys-sec. Tighten visitor mgmt.	Camera footage. Visitor logs. Badge access. Witness statements.
208	**Insider Recruitment / Bribery**	**Critical**	External actor recruits employee to provide access, data, or facilitate attack.	• TI report of insider recruitment forum ads targeting your company? • Employee reports outreach offering money for access?	• Recruitment confirmed? • Insider acted on it (data exfil, access granted)? • Scope of damage? • LE coordination?	• LE engage immediately • Legal hold + investigation • Revoke all access of insider	TP = recruitment + insider action. BP = caught early via report. FP = misunderstood outreach.	LE. Revoke. Confidential investigation.	Communication records (with consent / warrant). Account activity. Forensic image

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• Anomalous behaviour (sudden access patterns) of specific employee? • HR / ethics report?		• Brief leadership confidentially			of devices. Financial records.

Ransomware TTPs (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
209	**Shadow Copy Deletion (vssadmin)**	**Critical**	Attacker deletes Volume Shadow Copies to prevent recovery — strong ransomware precursor.	• vssadmin delete shadows / Resize ShadowStorage / Delete ShadowStorage? • wmic shadowcopy delete? • PowerShell removing shadows? • SYSTEM context running these?	• Hands-on attacker? • Encryption next? • Other hosts targeted? • Backup systems also targeted?	• IMMEDIATE — isolate host + activate ransomware IR • Block vssadmin delete via EDR • Hunt for active attacker on other hosts • Verify offline backups intact	TP = shadow delete from suspicious context. BP = blocked. FP = sanctioned IT (rare and concerning).	Isolate. IR. Verify backups.	EDR cmdline. Sysmon. Process tree. Pre-encryption timeline.
210	**Backup Targeting / Tampering**	**Critical**	Attacker accesses backup infrastructure to delete or encrypt backups before main encryption event.	• Access to backup server / appliance from non-backup-admin? • Mass delete events on backup repository? • Backup retention policies modified? • Backup admin account anomalous activity?	• Backup integrity? • Offline / immutable backups intact? • Coordinated with file encryption? • Recovery options remaining?	• Activate ransomware IR • Engage backup vendor • Verify offline / air-gapped backups • Plan recovery	TP = backup tamper + ransomware. BP = backup hardened. FP = sanctioned cleanup.	IR. Verify backups. Vendor.	Backup audit. Account history. EDR. Recovery viability.
211	**Encryption Stage Detection**	**Critical**	Active file encryption — mass file modifications with new extensions or encrypted headers.	• EDR alert for ransomware family (Lockbit, BlackCat, Royal, Akira)? • Files renamed with .encrypted / .locked / random extensions at scale? • Ransom note files appearing (README, HOW-TO)? • Multiple hosts encrypting simultaneously?	• Encryption family identification? • Spread mechanism? • Initial access vector? • Data exfil before encryption (double extortion)?	• IMMEDIATE — network isolate everything affected • Activate IR retainer • Coordinate with leadership, legal, possibly LE • Recover from clean backups	TP = encryption + ransom note. BP = blocked at execution. FP = legit encryption tool.	Isolate. IR. Recovery.	EDR timeline. Encrypted file samples. Ransom note. Initial access trace.
212	**Pre-Encryption Reconnaissance / Data Cataloguing**	**High**	Attackers map sensitive shares before encryption to maximise damage and exfiltrate for double extortion.	• Mass directory enumeration on file shares? • Tool indicators (ADRecon, FileMatic, copy of files in stage area)? • Many file-read events from one identity? • Exfil staging?	• Pre-encryption dwell time? • Data exfil already happened? • Hands-on attacker? • Imminent encryption?	• Activate IR — assume imminent encryption • Isolate identity / endpoints • Hunt for active attacker presence • Verify backups + plan recovery	TP = recon + ransomware operator pattern. BP = caught early. FP = legit IT survey.	Isolate. IR. Hunt.	File audit. EDR. Network exfil. Tool artefacts.
213	**Disable Recovery Features**	**Critical**	Attacker runs commands to disable Windows Recovery, BCD	• bcdedit /set recoveryenabled No? • wbadmin delete catalog -quiet?	• Hands-on attacker? • Encryption next? • Other hosts?	• IMMEDIATE — isolate + IR • Block these commands via EDR	TP = recovery disable + attacker. BP = blocked. FP =	Isolate. IR. Block.	EDR cmdline. Sysmon. Process tree.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
			edit, wbadmin delete catalog.	• bcdedit /set bootstatuspolicy ignoreallfailures? • SYSTEM context running these?	• Recovery still possible?	• Hunt fleet-wide • Plan recovery	none reasonable.		
214	**Mass Account Disruption (Domain-Wide)**	**Critical**	Just before encryption, attacker disables, locks, or modifies accounts to maximise impact.	• Mass 4720/4722/4725 events (account create/disable/lockout)? • Domain admin running these? • Coordinated timing across DCs? • User reports of mass logon failures?	• Hands-on attacker on DC? • Coordinated with encryption? • Recovery accounts also affected? • KRBTGT compromise?	• IMMEDIATE — disable affected admin accounts • Activate AD compromise IR • Recover accounts from backup AD • KRBTGT reset twice	TP = mass account disrupt + ransomware. BP = caught early. FP = none.	Disable admin. IR. AD recovery.	4720/4722/4725. DC logs. EDR on DCs. Account history.

Reverse Shells & C2 (6)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
215	**Cobalt Strike Beacon**	**Critical**	Cobalt Strike, the most common pen-test/attacker C2 framework — many TTPs and signatures.	• EDR signature for Cobalt Strike? • Beacon outbound at malleable-C2 cadence? • Named pipe (\\\\.\\pipe\\msagent_*) between processes? • Default port 50050 (team server) reached?	• Profile (default vs custom)? • Operator TTPs (lateral, kerberoast, recon)? • Persistence? • Hunt fleet-wide?	• Isolate hosts • Reimage • Reset all creds • Hunt for sister beacons + persistence	TP = beacon + operator TTPs. BP = blocked. FP = pen-test (verify).	Isolate. Reimage. Reset. Hunt.	EDR. Memory image. Network captures. Beacon config extraction.
216	**Sliver / Mythic / Brute Ratel C2**	**Critical**	Modern open-source / commercial C2 frameworks used by attackers (replacing Cobalt Strike).	• EDR signature for Sliver / Mythic / Brute Ratel? • HTTPS beaconing patterns matching framework? • Specific user-agents or JA3 fingerprints? • TI feed mentioning campaign?	• Operator TTPs? • Persistence mechanism? • Hands-on activity? • Hunt fleet-wide?	• Isolate • Reimage • Reset creds • Hunt + TI integration	TP = framework + ops. BP = blocked. FP = pen-test.	Isolate. Reimage. Hunt.	EDR. Memory. Network. JA3/JA4. Beacon config.
217	**Web Shell C2 (China Chopper, AntSword, Behinder)**	**Critical**	Web shell on compromised web server giving attacker RCE via HTTP(S).	• Suspicious file in web root (.aspx, .php, .jsp)? • HTTP request triggering process spawn from web user? • China Chopper / Behinder / Godzilla signatures? • Encrypted POST bodies?	• Shell capabilities? • Lateral from web server? • Persistence beyond shell? • Other web servers?	• Remove shells • Reimage / sanitise web server • Patch upload vector • Hunt for credentials taken	TP = shell + use. BP = upload blocked. FP = developer test (don't keep in prod).	Remove. Reimage. Patch. Hunt.	Web logs. File system. EDR. Shell sample. Network.
218	**DNS Tunnelling C2 (Cobalt Strike DNS, dnscat2)**	**High**	C2 traffic via DNS — covert channel.	• Volume of TXT / NULL / unusual record DNS to one domain? • Long random subdomains? • Traffic to authoritative server in volume? • TI domain reputation?	• Decoded payload? • Beacon cadence? • Volume implies data exfil? • Other hosts?	• Block domain at DNS • Force internal resolver • Isolate host • Reimage	TP = DNS tunnel + C2. BP = blocked. FP = legit DNS-based service.	Block. Isolate. Reimage.	DNS logs. EDR. Decoded payload. Domain WHOIS.
219	**HTTPS / TLS C2 with Domain Fronting**	**High**	C2 hides behind major CDN with SNI/Host mismatch (covered in network section but specific to C2).	• TLS proxy showing SNI/Host mismatch? • CDN egress from unusual app? • Beaconing pattern through CDN?	• Real backend? • Process initiating? • Other hosts? • Operator TTPs?	• Block backend • TLS inspection • Isolate • Reimage	TP = mismatch + C2. BP = blocked. FP = legit CDN-hosted app.	Block. Isolate. Reimage.	TLS proxy. NetFlow. EDR. Domain analysis.

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
				• TI flagging hidden domain?					
220	**Encrypted Outbound Tunnels (Tor, Lokibot, custom)**	**High**	Outbound to Tor relays or custom-encrypted C2 channels.	• Connection to known Tor entry nodes? • Custom encrypted protocol on non-standard port? • Beaconing to high-risk geo/ASN? • TI feed match?	• Process? • Persistence? • Operator TTPs? • Other hosts?	• Block Tor / suspicious destinations • Isolate • Reimage • Hunt	TP = Tor / custom C2 + ops. BP = blocked. FP = legit Tor research (rare).	Block. Isolate. Reimage.	Network. EDR. TI integration.

Cryptojacking (4)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
221	**Endpoint Cryptominer**	**Medium**	Mining malware on user endpoint — sustained CPU/GPU + outbound to mining pools.	• Sustained high CPU? • Mining pool DNS / connections? • Process names (xmrig, etc) or known mining hashes? • User performance complaints?	• Initial access? • Persistence? • Wallet address (TI)? • Spread to other hosts?	• Reimage • Block mining pools globally • Hunt • Patch root cause	TP = miner + pool. BP = blocked. FP = legit compute job.	Reimage. Block. Hunt.	EDR. Network. Wallet (TI). Persistence.
222	**Cloud Compute Hijack for Mining**	**High**	Attacker spins up large/many cloud instances using compromised creds for mining.	• Unexpected instance creation events (large types, GPU)? • Cost spike alert? • Outbound to mining pools from cloud? • IAM identity unusual?	• IAM compromise scope? • Resources hijacked + value? • Persistence? • Cloud-native or container-based?	• Terminate hijacked resources • Rotate creds • Audit IAM • Cloud provider credit	TP = unauthorised resource + mining. BP = SCP blocked. FP = legit workload.	Terminate. Rotate. Audit.	Cloud audit. IAM. Cost. Mining pool dest.
223	**Browser-Based Cryptomining (Coinhive-style)**	**Low**	Compromised website runs JS that mines cryptocurrency in visitor's browser.	• Visited site causing high CPU on user browser? • JS from cryptomining service (CoinHive successors)? • Many users visiting same site complaining? • Detection at proxy?	• Compromised site (own or third-party)? • Mining duration / impact? • Other affected sites? • Removal of script?	• Block mining JS at proxy / browser extension • If your site: clean JS + investigate • Brief users • Monitor	TP = mining JS + impact. BP = blocked. FP = legit script flagged.	Block. Clean if own. Monitor.	Web proxy. Browser logs. Site source.
224	**CI / Build System Mining**	**Medium**	Attacker abuses public CI runners (GitHub Actions, GitLab) or compromised pipelines for mining.	• CI job running unusually long? • Pipeline executing mining binary? • Public repo with malicious workflow? • Cloud cost from CI minutes?	• Source (compromised PR, supply chain)? • Mining duration? • Other repos? • Persistence?	• Cancel jobs • Revoke runner tokens • Audit pipelines • Restrict public CI	TP = CI mining. BP = blocked. FP = legit long-running job.	Cancel. Revoke. Restrict.	CI logs. Pipeline history. Network from runners.

Deception & Impersonation (5)

#	Attack	Severity	What it looks like	L1 Checks	L2 Checks	L3 / IR	TP / FP / BP	Containment	Forensics
225	**Typosquatting Domain**	**Medium**	Attacker registers domain similar to yours (g00gle.com, c0rp.com) for phishing or brand abuse.	• Brand monitoring or DNS feed alert? • Lookalike domain registered recently? • MX records configured (for phishing)? • Domain hosted with known abuse provider?	• Active phishing on domain? • Targeting your customers? • Coordinated campaign? • Take-down possible?	• Take-down request • Block domain at proxy + email • Notify customers if affected • Defensive registration of variants	TP = typosquat + abuse. BP = caught early. FP = legitimate similar name.	Block. Take-down. Notify.	Brand monitoring. DNS / WHOIS. Site content. Customer reports.
226	**Brand Impersonation (Phishing + Logo)**	**Medium**	Phishing site uses your logo, branding, copy to convince customers.	• Customer reports fake login site? • Anti-phishing service alert? • Search results for your brand showing fake? • Social media posts with fake link?	• Volume of victims? • Phishing kit (e.g. 16shop, EvilProxy)? • Attribution? • Take-down progress?	• Take-down via host / registrar • Notify customers + comms • Add to internal block-lists • Tighten brand protection	TP = brand abuse + phishing. BP = caught at monitoring. FP = legit similar branding.	Take-down. Customer comms.	Site contents. Phishing kit. Customer reports. Brand monitoring.
227	**Fake Vendor / Invoice Fraud**	**High**	Attacker poses as vendor sending fake invoice or asking to update payment details.	• Invoice with new bank details? • Vendor email matches but slightly off (lookalike)? • Pattern matches BEC playbook? • Finance flagged?	• Payment processed? • Real vendor compromised on their side? • Other targeted invoices? • Recovery possible?	• Halt payment + recover if possible • Verify with vendor independently • File with bank + LE • Implement vendor change verification process	TP = fake invoice + payment. BP = caught at verification. FP = legit vendor change.	Halt. Recover. LE.	Email + headers. Invoice. Vendor records. Bank records.
228	**Fake Job Offer / Recruiter Phishing**	**Medium**	Attacker poses as recruiter to phish current/former employees, or to recruit insiders.	• Employee reports unusual recruiter outreach? • Job offer with attachment / interview link? • LinkedIn fake recruiter? • Pretext too good to be true?	• Attachment / link malicious? • Coordinated targeting? • Insider recruitment angle? • Multiple employees?	• Brief employees • Take-down profiles • Hunt for related campaigns • Awareness training	TP = fake recruiter + phish/recruit. BP = identified. FP = legit recruiter.	Brief. Take-down.	LinkedIn / email records. Profile analysis. Subsequent attack correlation.
229	**Fake Mobile App (Counterfeit)**	**Medium**	Attacker publishes fake app on store mimicking your real app for cred theft / malware.	• App store search showing fakes? • Customer reports issues with 'app'? • App publisher not your org? • Malware behaviour in fake?	• Fake app capabilities (cred harvest, malware)? • Install count? • Take-down progress? • Customer impact?	• Take-down via store • Customer comms (use only official store/version) • Brand monitoring tightening • Hunt for variants	TP = counterfeit + abuse. BP = caught early. FP = legit similar app.	Take-down. Customer comms.	Store records. App package. Publisher records. Customer reports.