CYART — Red-Teaming Report (Week 4)

CYART — Red-Teaming Report (Week 4)
Deadline:04-11-2025
Author: Arbaz Shaikh

1. Engagement Overview

Scope: Full adversary simulation against a lab environment covering reconnaissance, cloud attack, phishing, C2, exfiltration and evasion.
Objective: Demonstrate end-to-end red-team campaign emulating real-world APT techniques; produce findings, detections observed, mitigation recommendations, and a PTES-style report for stakeholders.
Tools used: Kali Linux (Metasploit/msfvenom), Cobalt Strike (lab build), PoshC2 / PowerShell beacons, Pacu, awscli, CloudGoat, Caldera, Evilginx2, Metasploitable targets, Wazuh (blue team logging), Veil, proxychains, Tor, Mimikatz, PowerSploit, Draw.io, Google Docs.

2. Execution Summary (High-level)

- Recon: Performed network and cloud discovery (nmap, Pacu, awscli). Identified misconfigured S3 bucket with public read and an overprivileged IAM role.
- Initial Access: Phishing simulation with Evilginx2 to harvest credentials; successful credential capture for a mock web user.
- C2 Setup: Deployed Cobalt Strike HTTPS beacon and a stageless PowerShell beacon via PoshC2/PowerShell one-liner. Established a stable session to Windows VM.
- Privilege Escalation: Exploited IAM misconfiguration to escalate a cloud user to administrative privileges (lab simulation via Pacu).
- Lateral Movement & Persistence: Living-off-the-land via PowerShell and WMI; process injection into explorer.exe; persistence via scheduled tasks and startup registry keys.
- Exfiltration: Exfiltrated mock data from S3 to attacker-controlled cloud location and confirmed artifacts in logs.
- Evasion: Used msfvenom and Veil to obfuscate payloads, routed C2 through Tor with proxychains; fileless PowerShell execution to avoid AV.
- Detection: Wazuh generated alerts for suspicious access and privilege escalation; however some L.O.T.L actions bypassed signature-based detections.
- Reporting: Produced PTES-aligned report, attack path diagrams, and executive brief.

3. Detailed Steps (Reproducible, what you did)

3.1 Reconnaissance (Network & Cloud)

1. Network: nmap -sS -Pn -p- -T4 192.168.1.0/24 — enumerated live hosts and open services.
2. Cloud Recon: aws s3api list-buckets & aws s3 ls s3://<bucket> to enumerate S3 buckets; used Pacu module enum_s3 to validate bucket permissions.
3. Findings: Public-read S3 bucket containing mock data; discovered IAM user with overly permissive policies.

3.2 Initial Access (Phishing)

1. Built phishing site with Evilginx2 to proxy legitimate login and capture session tokens.
2. Sent simulated spear-phish to target user (lab account).

3. Captured session cookie and used it to authenticate to target web app, bypassing 2FA in lab simulation.

3.3 C2 Infrastructure & Payloads

1. Cobalt Strike (HTTPS beacon):
   - Generated a listener (HTTPS) with staging disabled for stealth (stageless PowerShell).
   - Configured teamserver and listener with valid-looking domain and TLS cert (lab-generated).
2. PoshC2 / PowerShell beacon: Generated one-liner stageless beacon and delivered via phishing payload.
3. Session Established:

Session ID	Target IP	Payload Type	Notes
SID001	192.168.1.50	PowerShell	Beacon established

C2 Setup (50-word summary):
Configured a Cobalt Strike HTTPS listener and deployed a stageless PowerShell beacon from PoshC2. The beacon used HTTPS for covert channeling; session SID001 connected to 192.168.1.50, providing an interactive remote shell and persistent control. Communication mimicked legitimate TLS traffic to reduce detection.

3.4 Privilege Escalation (Cloud)

- Used Pacu to probe IAM and discovered an over-privileged role attached to a service principal.
- Exploited policy attachment misconfiguration to assume role and obtain admin-level API keys in the lab.

Asset ID	Service	Misconfiguration	Notes
AID001	S3	Public read access	Vulnerable bucket

Privilege Escalation (50-word summary):
Pacu identified an over-privileged IAM role permitting role assumption. Using a chained API call exploit in the lab, the low-privilege user assumed the role and obtained admin-level keys, enabling arbitrary S3 access and further lateral movement into cloud services.

3.5 Lateral Movement and Living-Off-The-Land

- Used WMI and PowerShell (LOLBAS techniques) for lateral movement and remote command execution.
- Performed process injection into explorer.exe using PowerShell Invoke-ReflectivePE to hide payload.
- Used Mimikatz in a controlled lab to extract test credentials.

Attack ID	Tool	Action	Notes
LID001	PowerShell	Fileless execution	Bypassed AV

LOTL Lab (50-word summary):
Executed fileless PowerShell payloads leveraging native Windows tooling to run in-memory code and inject into explorer.exe. Harvested test credentials with Mimikatz and used WMI for remote command execution — demonstrating how living-off-the-land techniques reduce forensic artifacts and evade signature-based defenses.

3.6 Exfiltration
- Exfiltrated mock data from public S3 to attacker-controlled account using aws s3 cp. Confirmed transfer via logs and created decoy traffic to blend exfil.
3.7 Evasion & Obfuscation
- Encoded payloads with msfvenom and Veil to evade AV engines in lab.
- Routed traffic with proxychains through Tor to simulate anonymized C2 channels.

Payload ID	Type	AV Detection	Notes
PID001	Meterpreter	Bypassed	Obfuscated payload

Evasion (50-word summary):
Obfuscated Meterpreter payloads with Veil and msfvenom; routed C2 through Tor using proxychains to spoof source IPs. Fileless PowerShell and HTTPS tunnel minimized IOCs. Mixed decoy traffic with legitimate requests to make detection more challenging for network IDS.
4. Blue Team Observations & Detection Points
- Wazuh Alerts: Suspicious S3 access and privilege escalation triggers; alerts included abnormal API patterns and anomalous user agent strings.
- Gap 1 — LOTL Detection: Fileless PowerShell activity produced limited Windows Event IDs and left minimal disk artifacts — signature-based AV failed to detect several actions.
- Gap 2 — C2 over HTTPS: C2 masquerading as HTTPS blended with legitimate TLS, requiring TLS fingerprinting/JA3 or proxy inspection to identify anomalies.
- Gap 3 — Credential Replay: Evilginx2-style session hijacking bypassed typical password-only detection; session anomalies required behavioral detection.
5. Findings

Finding ID	TTP (MITRE)	CVSS Score	Remediation
FID001	Phishing (T1566)	7.5	Phishing-resistant MFA, user training
FID002	Cloud privilege misuse (T1588)	9.0	Least-privilege IAM, role-review, MFA for console/API
FID003	Fileless execution (T1059.001)	8.0	Endpoint detection for PowerShell logging, AMSI enforcement
FID004	C2 over HTTPS (T1071.001)	8.5	TLS inspection, JA3/SSL fingerprinting, Egress filtering

6. Recommendations (Concrete, prioritized)
Immediate (0–30 days)
- Enforce least-privilege IAM policies and run automated IAM policy audits.
- Enable CloudTrail + S3 object-level logging and guardrails for public buckets.
- Require phishing-resistant MFA (FIDO2 / hardware tokens) for privileged accounts.
- Enable PowerShell script block logging and AMSI; forward logs to SIEM.
Mid-term (30–90 days)
- Implement JA3/JA3S TLS fingerprinting and certificate pinning checks in network monitoring.
- Deploy behavioral analytics for session anomalies and lateral movement indicators.
- Harden endpoints with EDR tuned for in-memory/LOTL behaviors (behavioral rules).

- Run red-team exercises quarterly and adversary-emulation drills mapped to MITRE ATT&CK.

Long-term

- Adopt continuous validation of cloud infra with ScoutSuite/CloudCustodian and implement automated remediation for critical misconfigs.
- Integrate threat hunting program and tabletop exercises for exec-level incident response.

7. Key Learnings

1. C2 effectiveness: HTTPS beacons and stageless payloads yield robust control but require TLS inspection and behavior detection to uncover.
2. Cloud attack surface: Misconfigured S3 and over-privileged IAM roles are high-value, high-impact attack vectors — automation tools (Pacu/CloudGoat) quickly expose these.
3. LOTL danger: Native tools like PowerShell/WMI reduce detection noise; proper telemetry (script block logging, EDR) is essential.
4. Phishing sophistication: Session-hijacking proxies (Evilginx2) demonstrate the limits of password-only defenses and standard MFA.
5. Reporting value: PTES-style reporting + visual attack path diagrams are essential for making technical findings actionable for execs.

8. Outcomes- Short term: Reduction in successful phishing and credential-compromise incidents; improved detection of anomalous cloud API usage.

- Medium term: Significant decrease in privilege escalation opportunities; improved forensic trails from PowerShell/script execution.
- Long term: Organization attains proactive posture: continuous cloud compliance, faster detection and containment, reduced mean time to remediate (MTTR).

9. Artifacts Submitted

- Screenshots of Wazuh alerts and S3 object logs.
- Attack path diagram (Draw.io).
- PTES-compliant 200-word report and 100-word executive brief (separate documents).
- GitHub repo structure: cyart-red-teaming/Week 4/ containing Documentation, Screenshots, Workflows, Report.pdf, Readme.md.

10. Conclusion:

This report is built to be actionable: remediation steps are prioritized and mapped to findings.