Red Team Confluence Wiki

Core Concept

Red Team

A **Red Team** is a group of security professionals that simulate real-world adversaries to test an organization’s security posture. Unlike traditional security testing, Red Teaming is goal-oriented, often aiming to achieve objectives such as data exfiltration, domain dominance, or persistent access while avoiding detection.

Key Characteristics:

- **Adversary Emulation:** Mimics specific threat actors, their tactics, techniques, and procedures (TTPs).
- Full-Scope Testing: Includes social engineering, physical security, network exploitation, and more.
- Focus on Evasion: Red Teams attempt to bypass security controls and operate undetected.
- Real-World Attack Scenarios: Unlike vulnerability assessments or penetration tests, Red Teaming tests detection and response capabilities.

Red Team vs Penetration Test vs Vulnerability Assessment

Red Team Assessment: A goal-based adversarial simulation that emulates a real-world attack using the full spectrum of TTPs against all organizational attack surfaces (technical, physical, social) to test detection and response capabilities. It focuses on achieving specific objectives while avoiding detection.

Penetration Test: A focused technical assessment that identifies and exploits vulnerabilities in specific systems, networks, or applications to determine their exploitability and potential impact. It aims to find and validate as many vulnerabilities as possible within a defined scope.

Vulnerability Assessment: A systematic review to identify, classify, and prioritize vulnerabilities in systems, applications, and network infrastructure. It focuses on discovery and documentation without actually exploiting the vulnerabilities.

Feature	Red Teaming	Penetration Testing	Vulnerability Assessment
Objective	Simulate a real-world adversary attack	Identify and exploit security weaknesses	Identify vulnerabilities and misconfigurations
Scope	Broad, covers multiple attack vectors	Focused on specific systems/applications	Comprehensive review of vulnerabilities
Methodology	Adversary tactics, stealth, long-term persistence	Exploit known vulnerabilities to gain access	Identify and report vulnerabilities without exploitation
Testing Approach	Full-scope (physical, cyber, social engineering)	Controlled environment, usually black/gray box	Automated and manual scanning
Timeframe	Weeks to months	Days to weeks	Typically a short-term engagement
Stealth Required?	Yes, must avoid detection	No, detection not a primary concern	No, focuses on identification
Security Team Involvement	Tests Blue Team’s response & SOC capabilities	Security team may or may not be aware	Security team involved in patching
Deliverables	Executive report, technical findings, MITRE ATT&CK mapping	List of exploitable vulnerabilities, risk ratings	List of vulnerabilities, risk scores, recommendations
Best For	Testing an organization's full security maturity	Assessing security posture of specific assets	Continuous vulnerability management

OPSEC

OPSEC is a process that identifies critical information to determine if actions can be observed by adversaries, determines if information obtained by adversaries could be harmful, and then executes measures to eliminate or reduce vulnerabilities.

In red teaming, OPSEC refers to the practices and procedures used by the red team to protect their activities from detection by the blue team or other security monitoring systems. This includes:

- Infrastructure compartmentalization: Separating attack infrastructure to minimize correlation and attribution
- Communication security: Using encrypted and out-of-band channels for team communications
- Attribution obfuscation: Masking the true source of attacks
- Traffic patterns management: Ensuring red team activities mimic expected patterns or blend with normal traffic

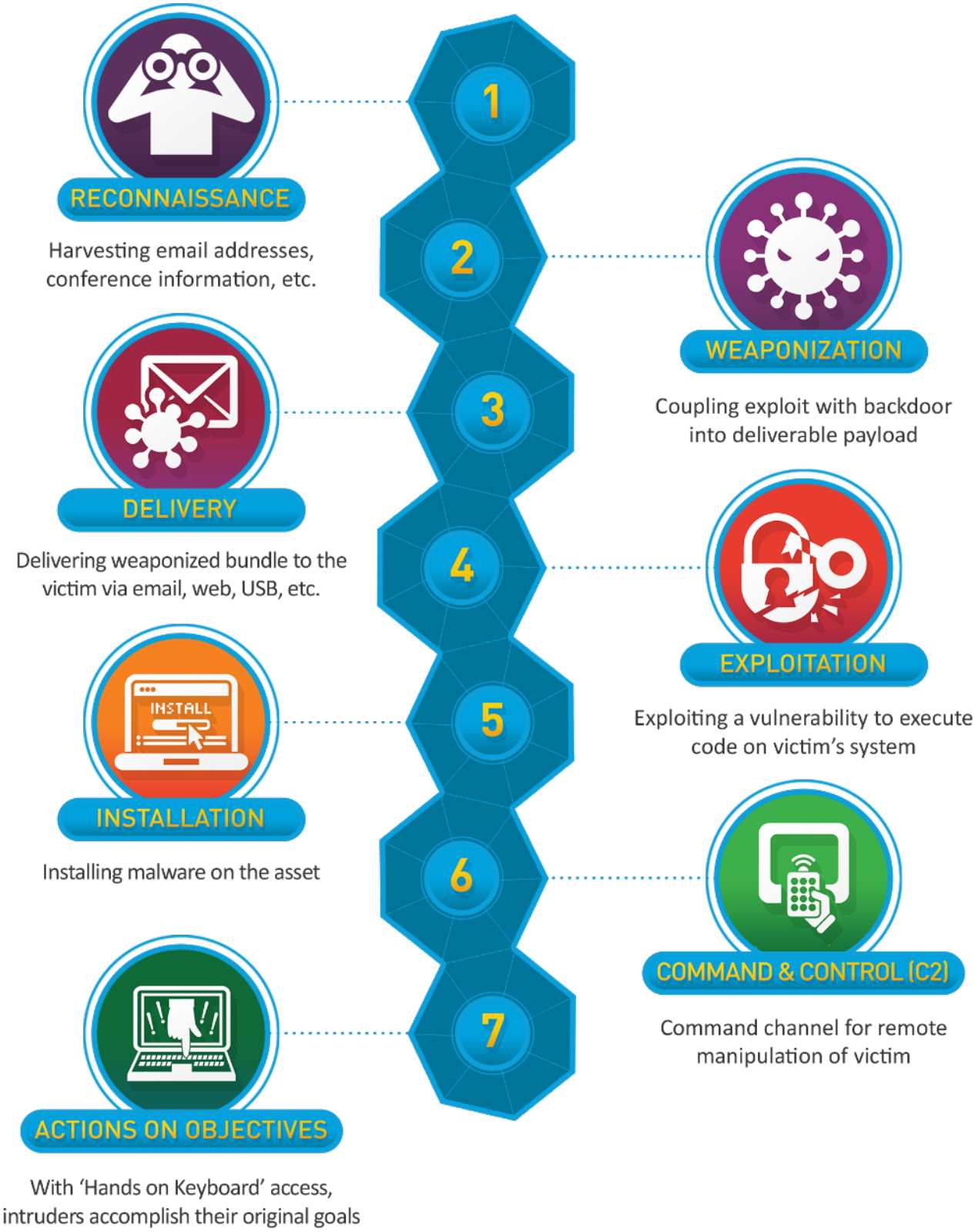
- Tool selection and modification: Using custom tools or modifying existing ones to avoid signature detection
- Operational tradecraft: Methodologies to minimize digital footprints and artifacts
- Data sanitization: Removing identifying metadata from files and communications

Proper OPSEC is crucial for red teams as premature detection can invalidate assessment results and fail to accurately test the organization's true detection capabilities.

Attack Life Cycle

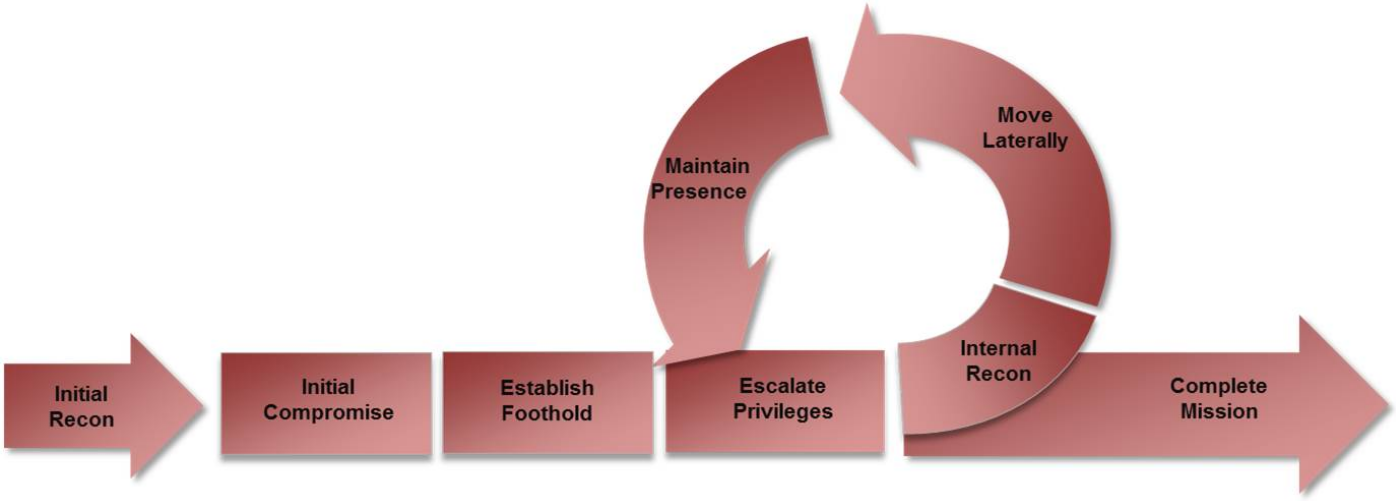
The attack lifecycle refers to the phases an adversary follows to achieve their objective, such as initial access, privilege escalation, lateral movement, and exfiltration. Various cybersecurity frameworks outline these steps:

1. Cyber Kill Chain (Lockheed Martin)



Attack life cycles are models that describe the sequence of steps attackers typically follow when compromising an organization.

Cyber Kill Chain (Lockheed Martin)



A 7-stage model describing the structure of an attack:

1. Reconnaissance: Gathering information about the target
2. Weaponization: Coupling exploits with backdoors into deliverable payloads
3. Delivery: Transmitting the weapon to the target environment
4. Exploitation: Triggering the attacker's code in the target environment
5. Installation: Installing malware or backdoor on the asset
6. Command & Control (C2): Establishing persistent remote control over the victim
7. Actions on Objectives: Executing the intended goals of the intrusion

Mandiant's Attack Lifecycle (now expanded to 8 phases)

Describes how targeted attacks unfold:

1. Initial Reconnaissance: Identifying targets and gathering intelligence
2. Initial Compromise: First breach of the target environment
3. Establish Foothold: Setting up persistent access
4. Escalate Privileges: Obtaining higher-level permissions
5. Internal Reconnaissance: Mapping the internal environment
6. Lateral Movement: Moving through the network to reach objectives
7. Maintain Presence: Ensuring continued access
8. Complete Mission: Achieving the attack objective (data exfiltration, destruction, etc.)

Red teams use these models to structure their activities and ensure their simulations accurately reflect real-world attack methodologies. They also provide a framework for organizations to understand where they need to implement defensive controls.

Engagement Planning

Engagement Planning forms the critical foundation of any successful red team operation. This comprehensive preparation phase ensures the exercise delivers meaningful security insights while

managing risks effectively.

The scope definition establishes clear boundaries for the assessment. Rather than simply listing systems as "in-scope" or "out-of-scope," proper scoping involves detailed discussions with stakeholders to understand business-critical assets, system interdependencies, and potential impact concerns. Technical boundaries must account for network segments, cloud environments, third-party integrations, and data flows. Physical location scoping requires consideration of access controls, sensitive areas, and safety concerns. The scope should also clearly articulate whether social engineering is permitted and which personnel groups may be targeted.

Threat modeling transforms the exercise from generic testing into a realistic simulation of relevant adversaries. Security teams analyze their organization's threat landscape to identify the most likely threat actors based on industry, geography, and data types. This involves researching actual TTPs employed by these adversaries, often leveraging intelligence reports and frameworks like MITRE ATT&CK. The red team can then map these capabilities against the organization's attack surface, prioritizing likely vectors and creating a campaign that mirrors real-world attacks the organization might face.

The breach model determines how the red team will establish initial access. This could range from a purely external assessment (starting with no access) to an assumed breach scenario (where some level of access is granted at the start). Organizations often select breach models that align with their most concerning threat scenarios. For example, a financial institution might focus on external breach scenarios, while a defense contractor might prioritize insider threat models. The breach model significantly impacts the engagement's timeline, required resources, and potential findings.

Notification and announcement strategies require careful balancing of operational realism against organizational risk. Full knowledge tests (where defenders know an exercise is occurring) sacrifice some realism but reduce business disruption risk. Limited knowledge tests restrict awareness to key personnel, while no-knowledge tests maximize realism but require robust emergency procedures. Most organizations implement a tiered notification approach with executives and key stakeholders aware of the general timeframe, while specific defensive teams remain uninformed. This approach requires developing communication plans for different scenarios, including potential business disruption events.

Rules of Engagement (ROE) serve as the authoritative governance document for the entire exercise. Beyond simply listing permitted techniques, comprehensive ROE define operational parameters including hours of operation, blackout periods (such as financial close periods or major business events), approval chains for high-risk activities, data handling protocols, and detailed escalation procedures. The ROE document should be treated as a legally binding agreement, signed by executive stakeholders, red team leadership, and legal representatives. It establishes liability boundaries and protections for both the organization and the red team members.

Record keeping and deconfliction processes prevent red team activities from causing unintended consequences. This includes maintaining detailed logs of all testing activities with timestamps, affected systems, techniques used, and results obtained. These records prove invaluable if an

incident occurs or if findings are questioned. Deconfliction mechanisms ensure red team activities don't conflict with legitimate security operations, other planned testing, or critical business functions. This typically involves establishing secure communication channels with a limited set of organizational contacts who can verify if observed anomalies are exercise-related.

Data handling frameworks address the potential exposure of sensitive information during testing. Red teams often encounter confidential data, intellectual property, or regulated information. Proper protocols define how such data should be documented (often using representative samples rather than actual data), how findings should be stored (usually encrypted and access-restricted), and how data should be securely destroyed post-engagement. These protocols must align with organizational compliance requirements and regulatory frameworks.

Duration planning extends beyond simply setting start and end dates. Effective timeline development involves mapping distinct phases (reconnaissance, initial access, privilege escalation, etc.) with realistic timeframes for each, incorporating buffer periods for unexpected challenges, and establishing clear milestones with stakeholder checkpoints. The duration should reflect the complexity of the environment and the sophistication of the simulated adversary, with advanced persistent threat simulations often spanning weeks or months to properly emulate realistic dwell times.

Resource allocation encompasses the people, technology, and infrastructure needed for success. Team composition should align with the required skill sets for the selected threat model, potentially including specialists in network penetration, social engineering, physical security, or specialized technologies. Technical resources include not only testing tools and software licenses but also infrastructure such as command and control servers, VPS hosting, domain registrations, and secure communication channels. Proper resource planning also addresses training needs if specialized skills are required for the engagement.

Post-Engagement and Reporting

Post-Engagement and Reporting transforms raw technical findings into meaningful security improvements. This phase elevates the exercise from a point-in-time test to a catalyst for organizational security maturation.

Evidence collection involves systematically gathering, organizing, and preserving all artifacts from the engagement. This includes not only screenshots of compromised systems but also tool outputs, network captures, command logs, system artifacts, and defensive alerts triggered. The evidence must maintain a clear chain of custody and be collected in a forensically sound manner. This comprehensive collection allows for detailed reconstruction of events and provides verification of findings if questions arise later.

Attack narratives translate technical details into compelling stories that illustrate security weaknesses. These narratives chronologically document the red team's journey, from initial access attempts through persistence, privilege escalation, lateral movement, data discovery, and

objective achievement. Effective narratives highlight not only successful techniques but also failed attempts and the process of discovery that led to success, providing defenders with insight into attacker methodology. By structuring these narratives around the ATT&CK framework, security teams gain context about how the observed behaviors relate to real-world threats and can better prioritize defensive improvements.

For example, rather than simply stating "The team exploited a vulnerable web application," a proper attack narrative would explain: "After discovering an outdated instance of Application X through passive reconnaissance, the team exploited CVE-2023-12345 to establish a foothold with limited user privileges. The team then identified misconfigured service accounts through local enumeration, leveraging these to escalate privileges and deploy a persistent backdoor that communicated through encrypted channels mimicking normal HTTPS traffic. This access enabled lateral movement to the financial database server through pass-the-hash techniques, ultimately extracting 250MB of simulated customer financial records over a three-day period without triggering existing monitoring systems."

Recommendations transcend simplistic vulnerability remediation to address systemic security gaps. Strategic recommendations focus on architectural improvements, security program enhancements, and long-term capability development. Tactical recommendations address specific vulnerabilities, misconfigurations, and technical controls. Procedural recommendations enhance detection capabilities, incident response workflows, and security operations. Effective recommendations provide clear implementation guidance with specific technologies, configuration changes, or process improvements rather than vague directives. Each recommendation includes a priority rating based on exploitation difficulty and potential impact, along with implementation complexity estimates and validation methods to confirm successful remediation.

Indicators of Compromise (IoCs) document the technical fingerprints left by the red team that mimic actual attacker artifacts. These include file hashes of tools and payloads, network indicators such as IP addresses and domain names used in command and control, host-based artifacts including registry modifications and file system changes, and process indicators such as command line parameters and service creations. These IoCs serve dual purposes: they allow the organization to verify if similar activities have occurred previously (indicating potential real compromises) and provide valuable detection content for security tools. Advanced red teams often develop custom YARA or Sigma rules alongside IoCs to enhance detection capabilities.

The reporting structure typically includes multiple documents tailored to different audiences. The executive summary translates technical findings into business risk terms, focusing on critical exposure areas, potential business impacts, and strategic recommendations. This document avoids technical jargon and focuses on governance, investment, and security program maturity. The technical report provides comprehensive details for security practitioners, including methodologies, tools, techniques, evidence, and detailed remediation steps. Many organizations also benefit from a remediation roadmap that sequences fixes based on risk, complexity, and dependencies, providing a practical implementation plan for addressing findings.

Debrief sessions facilitate knowledge transfer beyond written reports. Executive briefings focus on business risk and strategic improvements. Technical debriefs walk security teams through attack

methodologies and defensive failures, often including demonstrations of key techniques. Purple team sessions bring red and blue teams together to review detection gaps and improve monitoring capabilities. These interactive sessions allow defenders to ask questions, understand nuances, and develop deeper insight than reports alone can provide.

Remediation support extends the engagement's value through the improvement cycle. Rather than simply delivering findings and departing, effective red teams remain available during the remediation phase to clarify techniques, validate fixes, and provide technical guidance. Some organizations implement a phased verification approach where the red team retests specific findings after remediation to confirm effectiveness. This ongoing partnership ensures that security improvements actually address the underlying issues rather than implementing superficial fixes that attentive attackers could easily bypass.

TTP

TTPs are the patterns of activities and methods associated with specific threat actors or groups of threat actors. They represent how attackers operate and provide a framework for understanding, documenting, and communicating about attacker methodologies.

- Tactics: The high-level description of an attacker's objective or goal. Tactics represent the "why" of an attack technique (e.g., initial access, privilege escalation, lateral movement).
- Techniques: The specific methods used by adversaries to achieve tactical goals. Techniques represent the "how" of an attack (e.g., spear phishing, pass-the-hash, living off the land).
- Procedures: The detailed implementation of techniques. Procedures represent the exact steps, tools, and operational practices that adversaries use when executing techniques (e.g., specific malware variants, particular command sequences, custom scripts).

TTPs are important in red teaming for several reasons:

- They enable realistic emulation of specific threat actors relevant to the organization
- They provide a common language for describing attack methodologies
- They help organizations prioritize defenses based on actual attack patterns
- They allow for mapping of defensive controls to specific adversary behaviors

Red teams select and implement TTPs based on threat intelligence about adversaries targeting the organization's industry or geographic region, creating more realistic and valuable security assessments.

ATT&CK

MITRE ATT&CK is a globally-accessible knowledge base and framework that catalogs adversary tactics and techniques based on real-world observations. It serves as a comprehensive, structured representation of attacker behaviors, spanning the entire attack lifecycle.

Key characteristics of the ATT&CK framework:

- Structure: Organized hierarchically into Tactics (categories of technical objectives), Techniques (methods to achieve tactical goals), and Sub-techniques (specific implementations of techniques)
- Matrices: Different matrices for various environments:
  - Enterprise (Windows, macOS, Linux)
  - Mobile (iOS, Android)
  - ICS (Industrial Control Systems)
  - Cloud (AWS, Azure, GCP, SaaS)
- Additional Components:
  - Groups: Known threat actors and their associated TTPs
  - Software: Tools, malware, and utilities used by threat actors
  - Mitigations: Defensive measures mapped to specific techniques
  - Data Sources: Telemetry types useful for detecting techniques
- Use in Red Teaming:
  - Provides a common vocabulary for describing attack behaviors
  - Enables creation of threat-informed scenarios based on real adversaries
  - Facilitates documentation of testing coverage and gaps
  - Allows mapping of defensive capabilities to specific attack techniques
  - Supports reporting that connects findings to real-world threat behaviors

ATT&CK has become the de facto standard for describing adversary behavior in the security industry. Red teams use it to plan, execute, and document their operations, ensuring assessments are grounded in real-world attack methodologies and providing organizations with actionable intelligence about their security posture relative to actual threats.

External Reconnaissance

External reconnaissance represents the critical intelligence-gathering phase where threat actors (and red teams emulating them) collect information about target organizations without directly engaging their systems. This phase establishes the foundation for subsequent attack stages by mapping the attack surface, identifying potential vulnerabilities, and gathering intelligence on organizational structure and personnel.

It is typically divided into two categories:

1. Passive Reconnaissance (OSINT)
2. Active Reconnaissance

OSINT

OSINT refers to the collection and analysis of publicly available information to support reconnaissance. For Red Teams, OSINT provides a low-risk, passive method to map a target’s digital footprint, infrastructure, and personnel without alerting defenses.

DNS

Domain Name System records provide valuable insights into an organization's digital infrastructure. Attackers analyze DNS records to identify subdomains, IP address ranges, mail servers, and third-party service integrations. Tools like DNSdumpster, SecurityTrails, and DNSrecon allow systematic enumeration of these records, revealing potential entry points and the overall network topology. For example, discovering a forgotten subdomain pointing to legacy infrastructure often reveals vulnerable systems not maintained to current security standards.

Key Info:

- MX (Mail Exchange) records – potential phishing targets.
- TXT records (e.g., SPF, DKIM, DMARC) – email security configurations.
- NS (Name Server) records – potential misconfigurations.
- PTR (Reverse DNS) – possible internal naming conventions.

Tools & Methods:

- `nslookup` (Windows/Linux)
- `dig` (Linux/macOS)
- `host` (Linux/macOS)
- Online services like SecurityTrails, DNSDumpster, and VirusTotal.

whois

WHOIS database queries reveal domain registration details, including registration dates, expiration information, registrar data, and sometimes administrative contact information. This data helps establish the organization's digital history, identify acquisition targets through historical ownership changes, and potentially discover contact information for phishing campaigns. WHOIS information can also reveal related domains through common registrant patterns, expanding the potential attack surface.

Key Info:

- Organization name.
- Registrant contact details.
- Registrar information.
- Domain creation/expiration date.

Tools & Methods:

- `whois <domain>` (Linux/macOS).
- Online lookup services: **WhoisXML API**, **Whoisology**, **ICANN Whois**.

Social Media

Professional networking sites like LinkedIn provide detailed organizational structures, employee roles, technologies used, and recent hiring trends. Instagram, Twitter, and Facebook often contain unintentional disclosures such as badges, workplace photos, or system information visible in backgrounds. Red teams systematically map key personnel, focusing on technical staff, security teams, and executives for potential targeting. Social media also reveals organizational relationships, third-party vendors, and potential trust relationships that might be exploited.

Key Info:

- Employee names, roles, and email formats.
- Organizational culture and technology stack clues.
- Potential phishing pretexts.

Common Platforms & Targets:

- **LinkedIn** – Employee job titles, connections, company hierarchy.
- **Instagram/Facebook/Twitter (X)** – Company events, office locations.
- **GitHub** – Public repositories, exposed API keys, internal development leaks.

Tools & Methods:

- **theHarvester** – Gathers emails, subdomains, and names from public sources.
- **Maigret** – Collects user profiles from various social platforms.
- **GHunt** – Gathers intelligence on Google accounts.

Official Website

Corporate websites contain extensive information valuable to attackers. Career pages disclose technologies used internally through job requirements. Contact pages reveal office locations,

phone systems, and email naming conventions. Press releases and news sections highlight acquisitions, partnerships, and major IT initiatives. Sitemaps expose content structures and potentially restricted areas. Red teams methodically scrape and analyze this content, building targeted attack scenarios based on specific organizational details rather than generic approaches.

Key Info:

- **Contact pages** – Extract employee email formats.
- **Career pages** – Identify tech stack from job descriptions.
- **Sitemap.xml** – Find hidden sections of the website.
- **Press/News** – Identify partnerships, security incidents, or upcoming changes.

Tools & Methods:

- **Burp Suite Spider** – Crawls websites to map directories.
- **Google Chrome Developer Tools** – Inspect network requests and hidden endpoints.
- **robots.txt** – Checks for disallowed paths.

Passive Subdomain Enumeration

Discovering subdomains without directly interacting with target infrastructure relies on certificate transparency logs, passive DNS databases, and search engine cached results. Tools like Amass, Subfinder, and CertSpotter aggregate data from dozens of sources to build comprehensive subdomain maps. This reveals development environments, testing platforms, API endpoints, and other specialized infrastructure that may have different security postures than main corporate systems. Each subdomain represents a potential entry point with potentially different security controls.

Key Info:

- Forgotten or misconfigured services.
- Internal applications exposed to the public.
- Staging/testing environments.

Tools & Methods:

- **crt.sh** – Extracts SSL/TLS certificate-related subdomains.
- **subfinder** – Aggregates subdomains from multiple passive sources.
- **Amass (Passive Mode)** – Collects subdomains without actively probing.

Dorking

Search engine operators and specialized syntax allow precise filtering of publicly indexed information about target organizations. Google dorking uses advanced operators like "site:", "filetype:", and "intext:" to discover exposed documents, configuration files, and sensitive data. Shodan dorking identifies Internet-facing devices, control systems, and unusual services associated with the organization. GitHub dorking can reveal leaked credentials, API keys, and internal code that exposes architectural vulnerabilities. These techniques often reveal information organizations don't realize is publicly accessible.

Google Dorking

- Finds exposed files, login portals, or misconfigured services.

Shodan Dorking

- Identifies internet-exposed services and devices.

GitHub Dorking

- Finds sensitive credentials, API keys, and hardcoded secrets.

Tools for Automating Dorking:

- **Google Hacking Database (GHDB)** – List of useful Google dorks.
- **Shodan CLI** – `shodan search`
- **Gitrob / TruffleHog** – Finds sensitive data in GitHub repositories.

Active Reconnaissance

Active reconnaissance involves direct interaction with target systems, generating network traffic and potentially triggering security monitoring. These techniques provide detailed technical information but carry a higher risk of detection.

Port Scan

Port scanning systematically probes target IP ranges to identify open ports, running services, software versions, and operating systems. Complex scanning strategies balance detection avoidance against comprehensive coverage. Full port scans (all 65,535 TCP/UDP ports) identify unusual services and non-standard configurations. Version scanning determines specific software and firmware versions, enabling precise vulnerability mapping. Red teams typically employ distributed scanning, timing alterations, and decoy techniques to avoid triggering defensive alerts

while building detailed service maps of the target environment.

Port	Service
21	FTP
22	SSH
25	SMTP
53	DNS
80	HTTP
443	HTTPS
3389	RDP

Tools & Methods:

- **Nmap** – `nmap -sS -p- -A example.com`
- **Masscan** – Ultra-fast scanning of large IP ranges.
- **RustScan** – Faster alternative to Nmap for initial scans.

Directory Bruteforce

Directory brute forcing attempts to discover hidden or unlinked content on web servers by systematically testing thousands of potential directory and file names. This technique often reveals backup files, administrative interfaces, development resources, and improperly secured content. Tools like Gobuster, Dirsearch, and FFUF combine common wordlists with target-specific terms (company names, products, etc.) to identify valuable resources. Discovered endpoints are then analyzed for vulnerabilities, improper access controls, or information leakage that could facilitate further compromise.

Common Findings:

- **/admin/** – Admin portals.
- **/backup/** – Backup files.
- **/test/** – Staging/test environments.
- **/.git/** – Exposed Git repositories.

Tools & Methods:

- **Dirb** – `dirb http://example.com /path/to/wordlist.txt`
- **Gobuster** – `gobuster dir -u http://example.com -w common.txt`
- **FFUF** – Fast fuzzing and enumeration.

Custom Wordlists for Brute Forcing

- **SecLists** – `/usr/share/seclists/Discovery/Web-Content/`
- **Raft Wordlists** – `/path/to/raft-large-files.txt`
- **Common Crawl Data** – Extracted from large-scale web scraping datasets.

Vulnerability Scanning

Vulnerability scanning in red team operations focuses on identifying exploitable weaknesses in external-facing systems while maintaining operational security. Unlike standard security scanning, red team vulnerability scanning employs techniques designed to minimize detection while still gathering actionable intelligence.

During external reconnaissance, vulnerability scanning examines discovered systems to determine specific software versions, patch levels, and security weaknesses. Red teams typically limit scans to high-value targets identified through passive methods, rather than conducting comprehensive scans that generate significant traffic.

To avoid detection, red teams distribute scanning activity over time, use multiple source points, and carefully time scans to blend with normal network traffic patterns. Custom scan configurations focus solely on externally exploitable vulnerabilities relevant to gaining initial access, ignoring internal issues that would be irrelevant at this stage.

By employing these measured approaches, red teams can identify potential entry points while minimizing defensive alerts. This balance between thoroughness and stealth reflects the methodical approach used by sophisticated adversaries who may spend weeks or months in the reconnaissance phase before attempting exploitation.

1 Passive Vulnerability Enumeration (Low OPSEC Risk)

- **Extracts vulnerability data without direct interaction.**
- Useful for avoiding detection while still gathering useful information.

2 Active Vulnerability Scanning (High OPSEC Risk)

- **Direct interaction with target systems to identify exploitable weaknesses.**
- Requires careful **rate limiting and obfuscation** to avoid detection.

Tools & Methods:

- Service Version Information gathering
- Nuclei
- Nessus
- Burpsuite

Initial Access

Initial access techniques represent the methods adversaries (and red teams emulating them) use to gain their first foothold within a target environment. These techniques focus on breaching the network perimeter or obtaining initial system access, forming the foundation for all subsequent attack activities.

Password-based Attack

Password-based attacks exploit weak or compromised authentication credentials to gain legitimate access to systems. Rather than exploiting technical vulnerabilities, these attacks target human tendencies and identity management weaknesses.

Password spraying uses a small set of common passwords against many different accounts, avoiding account lockouts by limiting attempts per account. This approach takes advantage of organizational password policies that often allow at least some users to select predictable passwords. Red teams typically target exposed services like VPN portals, email access, or remote work platforms using passwords based on seasons, company names, or common patterns.

Brute-force attacks attempt all possible password combinations against specific high-value accounts. Modern defenses largely mitigate traditional brute-force attacks, but they remain effective against offline password hashes, misconfigured services, or legacy systems lacking proper lockout policies.

Educated-guess attacks leverage target-specific information gathered during reconnaissance to construct password lists. This might include company terminology, sports teams from the organization's location, or permutations of known naming conventions. This targeted approach increases success probability while generating less suspicious activity than broader attacks.

Leveraging leaked passwords involves using credentials from public data breaches to attempt access to corporate systems. This attack vector exploits password reuse across personal and professional accounts. Red teams cross-reference email addresses found during reconnaissance with breach databases to obtain potential passwords for testing.

Attack Type	Description	OPSEC Risk
Password Spraying	Attempts **one or a few passwords across many accounts** to avoid account lockouts.	**Medium** (Detectable with failed login correlation)
Brute Force	Systematically tries all possible password combinations.	**High** (Triggers account lockouts, SIEM alerts)
Credential Stuffing	Uses leaked username-password pairs from **data breaches**.	⚠ **Medium-High** (Unusual login behavior detection)

Attack Type	Description	OPSEC Risk
Educated Guessing	Uses **personal or company-related information** (e.g., `CompanyName2024!`).	⚠ **Medium** (May bypass simple security policies)
Leaked Passwords	Searches for previously exposed credentials from **data dumps** (e.g., **HaveIBeenPwned, Dehashed, BreachCompilation**).	**Low** (Passive reconnaissance method)

Common Targets

- **Enterprise login portals** (VPNs, Citrix, OWA, SSO platforms).
- **Cloud platforms** (**AWS, Azure, Google Cloud**).
- **Remote Access Services** (**RDP, SSH, VNC**).

OPSEC Considerations

Use slow and distributed attacks to avoid triggering lockouts.
Utilize compromised proxy networks to obfuscate attack origin.
Use user-agent rotation to mimic real user logins.

Phishing

Phishing attacks use social engineering to trick users into providing access or executing malicious code, often representing the path of least resistance into otherwise well-secured environments.

Adversary-in-the-Middle (AitM) phishing creates convincing replicas of legitimate authentication portals, intercepting credentials or authentication tokens when users attempt to log in. Modern AitM techniques can bypass multi-factor authentication by capturing and replaying tokens in real-time sessions. Red teams deploy these attacks through targeted emails directing users to carefully crafted lookalike domains.

Malicious attachment phishing delivers weaponized documents or files that exploit application vulnerabilities or execute malicious code when opened. Red teams craft these attachments to evade security scanning, using techniques like macro obfuscation, living-off-the-land binaries, or recently discovered exploits. These attachments typically establish remote access tools or download additional malware stages.

Contextual phishing customizes attack narratives based on organizational events or user responsibilities discovered during reconnaissance. By referencing actual projects, colleagues, or timely concerns, these targeted phishing attempts achieve significantly higher success rates than generic campaigns.

Types of Phishing Attacks

Attack Type	Description	Common OPSEC Risks
Credential Phishing	Fake login pages to steal usernames/passwords.	**High** (If using corporate infrastructure)
Adversary-in-the-Middle (AiTM) Phishing	Uses **reverse proxies (Evilginx, Modlishka)** to bypass MFA.	**High** (Triggers MFA push notifications)
Malicious Attachment Phishing	Sends infected Office docs, PDFs, or LNK files with malware payloads.	⚠ **Medium** (Detections in email security gateways)
Vishing (Voice Phishing)	Phone-based social engineering to extract login details.	**Low** (Less digital footprint)

OPSEC Considerations

Use custom phishing kits instead of widely known tools.
Host payloads on compromised sites instead of newly registered domains.
Leverage trusted email senders (e.g., **compromised business email accounts**).

Configure SPF, DKIM, DMARC

Tools for Red Teaming

- **Evilginx2** – AiTM phishing for capturing MFA sessions.
- **GoPhish** – Open-source phishing campaign manager.
- **Modlishka** – Real-time credential proxying.

Exploit Public-facing Service

Exploiting public-facing services takes advantage of technical vulnerabilities or misconfigurations in Internet-exposed systems to gain unauthorized access without requiring user interaction.

Vulnerability exploitation involves leveraging known or zero-day security flaws in web applications, VPN services, email servers, or other external systems. Red teams prioritize recently disclosed vulnerabilities that organizations may not have patched, particularly those affecting widely used platforms or those known to exist in the target environment based on reconnaissance findings.

Misconfiguration exploitation targets improperly secured systems rather than software flaws. This includes default credentials, unnecessary service exposure, excessive permissions, or insecure configuration options. Red teams systematically test discovered services for common misconfigurations such as exposed administrative interfaces, insecure authentication methods, or overly permissive API endpoints.

Common Targets

Service	Common Vulnerabilities

Web Applications	**SQL Injection (SQLi), Remote Code Execution (RCE), Local File Inclusion (LFI), SSRF**
Remote Access Services	**RDP, VPNs, Citrix, SSH misconfigurations**
Email & Collaboration	**Microsoft Exchange (ProxyShell, ProxyLogon), Atlassian Confluence (RCE bugs)**
Cloud Services	**Exposed S3 Buckets, Open Elasticsearch, Misconfigured IAM Roles**

□□ Exploitation Methods

- **Finding Unpatched CVEs** – Identifying outdated services with **public exploits**.
- **Misconfiguration Attacks** – Exploiting **default credentials, open APIs, or weak ACLs**.
- **Zero-day Exploits** – Using **undisclosed vulnerabilities** to gain access.

□□ OPSEC Considerations

□**Use Tor/VPNs or compromised infrastructure** to launch exploits.
□**Test for rate-limiting and WAF behavior before launching RCE attacks.**
□**Use crafted payloads with encrypted shells (e.g., Cobalt Strike, Meterpreter).**

Valid Account

Valid account techniques use legitimate credential sets to access systems normally, generating minimal suspicious logging while providing the same access as authorized users.

Leaked credentials from third-party breaches provide ready-made access when users reuse passwords across services. Red teams collect and test credentials from public breach databases, focusing on email addresses associated with the target organization.

Credentials sold on black markets sometimes include access to corporate systems directly, particularly for compromised managed service providers or third-party vendors. While ethical red teams don't purchase such credentials, they may simulate this vector by using discovered or provided test accounts to represent this increasingly common attack path.

How Attackers Obtain Valid Accounts

Method	Description

Leaked Credentials	**Checking breach databases** (Dehashed, LeakIX, HaveIBeenPwned).
Black Market Accounts	Buying **compromised corporate accounts** from underground forums.
Previous Breach Reuse	Using **old breach data** to log into **new services**.
Business Email Compromise (BEC)	**Hijacking** an employee's email to pivot internally.

OPSEC Considerations

- **Use residential proxies to mimic real geographic locations.**
- **Test logins slowly to avoid triggering alerts (smart timing).**
- **Blend into normal user activity (don't perform immediate privilege escalation).**

Supply Chain Pollution

Supply chain compromise involves attacking the target indirectly by first compromising a trusted vendor, software provider, or service that has privileged access to the primary target.

Third-party provider compromise exploits trust relationships with vendors who have legitimate access to the target environment. Red teams identify key service providers during reconnaissance and may simulate compromise of these providers to test how the organization handles third-party risks.

Software distribution compromise simulates attacks where legitimate software updates or packages are replaced with malicious versions. Red teams may test whether organizations verify the integrity of updates, particularly for critical operational software.

Development pipeline pollution represents sophisticated attacks targeting source code repositories, build systems, or deployment pipelines. By inserting malicious code early in the development process, attackers can ensure their code is distributed through legitimate channels. Red teams may test for proper code signing, review processes, and integrity verification in the software supply chain.

Common Supply Chain Attack Vectors

Method	Description	Notable Attacks
Malicious Software Updates	Injecting **backdoors into trusted software** updates.	**SolarWinds (SUNBURST backdoor)**
Compromised Third-party Services	Hijacking **SaaS platforms** to target multiple clients.	**3CX Supply Chain Hack**
Backdoored Open-source Libraries	Uploading **malicious NPM/PyPI packages** to infect developers.	**Color.js & faker.js incident**

Method	Description	Notable Attacks
Hardware Tampering	Modifying **devices before deployment** to include spyware.	**Supermicro server espionage rumors**

□□ OPSEC Considerations

□Target less monitored third-party services.
□Use dormant implants to delay immediate detection.
□Leverage CI/CD pipelines for long-term persistence.

Revision #6
Created 23 March 2025 23:48:27 by winslow
Updated 24 March 2025 03:48:22 by winslow