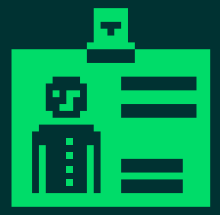
Defendable

Penetration Testing

17 November 2025



Anders Lund

Jobbet innen sikkerhet siden 2007

- Software developer
- Reverse Engineering av malware
- Sårbarhetsforskning
- Penetration Testing

Offensive Security Certified Professional

Offensive Security Certified Expert

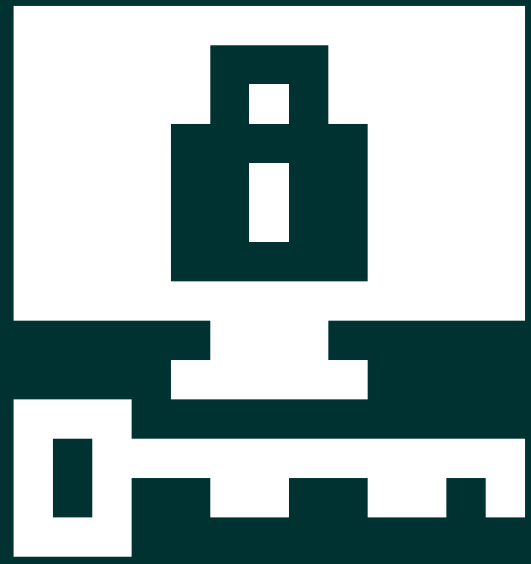
Offensive Security Web Expert

Offensive Security Exploitation Expert

Agenda

- Hva er penetrasjonstesting og Red Team øvelser
- Hvorfor utfører vi penetrasjonstesting og Red Team øvelser
- Hvordan utføres penetrasjonstesting
  - OWASP
  - MITRE ATT&CK
- Real World eksempel på Red Team øvelse

Penetration Testing



Hva er penetrasjonstesting

Simulert angrep på et system eller applikasjon for å identifisere og utnytte sårbarheter

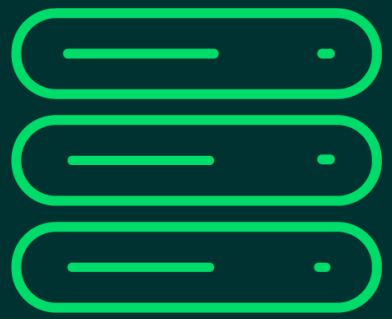
Hjelper organisasjoner å forstå:

- Hvor svakheter eksisterer
- Hvordan disse kan kobles sammen og utnyttes
- Hva er reelle konsekvenser hvis sårbarhetene utnyttes

Hva er en Red Team øvelse

- En red team øvelse er en effektiv demonstrasjon av konkret risiko som utgjøres av en APT (Advanced Persistent Threat).
- Testere vil imitere ekte angrep utført av avanserte trusselaktører ved å forsøke å kompromittere forhåndsbestemte mål gjennom kjente taktikker, teknikker og prosedyrer (TTP) og unngå å bli oppdaget.
- Disse omfattende, komplekse sikkerhetstestene er best egnet for virksomheter som ønsker å forbedre en moden sikkerhetsorganisasjon.

Elementer som testes i en Red Team øvelse

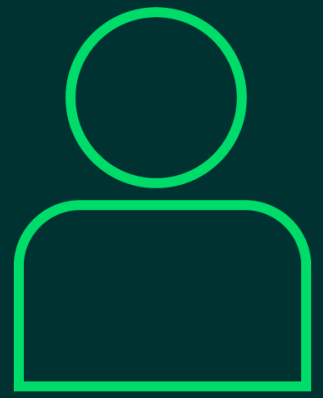


Teknologi

Internett-eksponerte tjenester og interne IT-nettverk
Skymiljøer

Fysisk

Bygninger, kontorer og fysisk infrastruktur



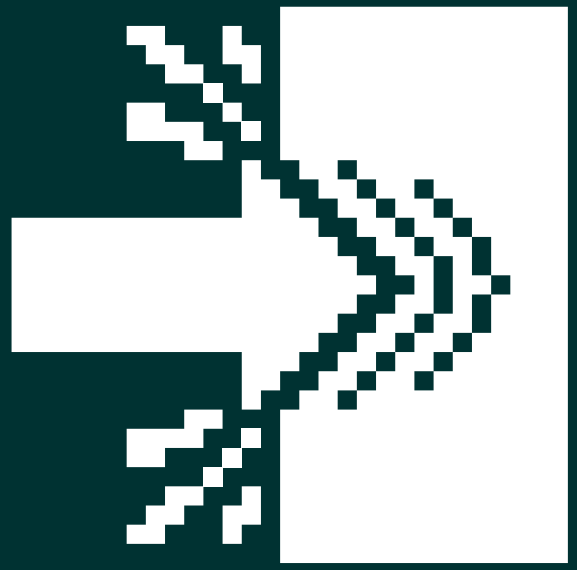
Mennesker og prosesser

Ansatte
IT-sikkerhet (Blue team / SOC)
Leverandører

Hvorfor penetrasjonstesting

- Avdekke reelle svakheter som angripere faktisk kan utnytte – ikke bare automatisk sårbarhetsskanning
- Forstå forretningskonsekvensene av mulige sikkerhetsbrudd.
- Teste evnen til å oppdage og reagere, ikke bare tekniske feil.
- Støtte kontinuerlig forbedring av sikkerhet gjennom hele utviklingsløpet.

Hvordan tester vi



OWASP TOP 10

- Dekker de vanligste og mest alvorlige web sårbarheter
- Oppdateres jevnlig basert på sikkerhetstrender
- Benyttes i penetrasjonstesting og sikker utvikling
- Penetrasjonstestere bruker OWASP Top 10 som et rammeverk for applikasjonstesting
- Hver kategori representerer en mulig angrepsvektor

OWASP TOP 10

A01:2021-Broken Access Control
A02:2021-Cryptographic Failures
A03:2021-Injection
A04:2021-Insecure Design
A05:2021-Security Misconfiguration
A06:2021-Vulnerable and Outdated Components
A07:2021-Identification and Authentication Failures
A08:2021-Software and Data Integrity Failures
A09:2021-Security Logging and Monitoring Failures
A10:2021-Server-Side Request Forgery


MITRE
ATT&CK™

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
Drive-by Compromise	AppleScript	.bash_profile and .bashrc	Access Token Manipulation	Access Token Manipulation	Account Manipulation	Account Discovery	AppleScript	Audio Capture	Commonly Used Port	Automated Exfiltration	Data Destruction
Exploit Public-Facing Application	CMSTP	Accessibility Features	Accessibility Features	Binary Padding	Bash History	Application Window Discovery	Application Deployment Software	Automated Collection	Communication Through Removable Media	Data Compressed	Data Encrypted for Impact
External Remote Services	Command-Line Interface	Account Manipulation	AppCert DLLs	BITS Jobs	Brute Force	Browser Bookmark Discovery	Distributed Component Object Model	Clipboard Data	Connection Proxy	Data Encrypted	Defacement
Hardware Additions	Compiled HTML File	AppCert DLLs	AppInit DLLs	Bypass User Account Control	Credential Dumping	Domain Trust Discovery	Exploitation of Remote Services	Data from Information Repositories	Custom Command and Control Protocol	Data Transfer Size Limits	Disk Content Wipe
Replication Through Removable Media	Control Panel Items	AppInit DLLs	Application Shimming	Clear Command History	Credentials in Files	File and Directory Discovery	Logon Scripts	Data from Local System	Custom Cryptographic Protocol	Exfiltration Over Alternative Protocol	Disk Structure Wipe
Spearphishing Attachment	Dynamic Data Exchange	Application Shimming	Bypass User Account Control	CMSTP	Credentials in Registry	Network Service Scanning	Pass the Hash	Data from Network Shared Drive	Data Encoding	Exfiltration Over Command and Control Channel	Endpoint Denial of Service
Spearphishing Link	Execution through API	Authentication Package	DLL Search Order Hijacking	Code Signing	Exploitation for Credential Access	Network Share Discovery	Pass the Ticket	Data from Removable Media	Data Obfuscation	Exfiltration Over Other Network Medium	Firmware Corruption
Spearphishing via Service	Execution through Module Load	BITS Jobs	Dylib Hijacking	Compile After Delivery	Forced Authentication	Network Sniffing	Remote Desktop Protocol	Data Staged	Domain Fronting	Exfiltration Over Physical Medium	Inhibit System Recovery
Supply Chain Compromise	Exploitation for Client Execution	Bootkit	Exploitation for Privilege Escalation	Compiled HTML File	Hooking	Password Policy Discovery	Remote File Copy	Email Collection	Domain Generation Algorithms	Scheduled Transfer	Network Denial of Service
Trusted Relationship	Graphical User Interface	Browser Extensions	Extra Window Memory Injection	Component Firmware	Input Capture	Peripheral Device Discovery	Remote Services	Input Capture	Fallback Channels		Resource Hijacking
Valid Accounts	InstallUtil	Change Default File Association	File System Permissions Weakness	Component Object Model Hijacking	Input Prompt	Permission Groups Discovery	Replication Through Removable Media	Man in the Browser	Multi-hop Proxy		Runtime Data Manipulation
	Launchctl	Component Firmware	Hooking	Control Panel Items	Kerberoasting	Process Discovery	Shared Webroot	Screen Capture	Multi-Stage Channels		Service Stop
	Local Job Scheduling	Component Object Model Hijacking	Image File Execution Options Injection	DCShadow	Keychain	Query Registry	SSH Hijacking	Video Capture	Multiband Communication		Stored Data Manipulation
	LSASS Driver	Create Account	Launch Daemon	Deobfuscate/Decode Files or Information	LLMNR/NBT-NS Poisoning and Relay	Remote System Discovery	Taint Shared Content		Multilayer Encryption		Transmitted Data Manipulation
	Mshta	DLL Search Order Hijacking	New Service	Disabling Security Tools	Network Sniffing	Security Software Discovery	Third-party Software		Port Knocking		
	PowerShell	Dylib Hijacking	Path Interception	DLL Search Order Hijacking	Password Filter DLL	System Information Discovery	Windows Admin Shares		Remote Access Tools		
	Regsvcs/Regasm	External Remote Services	Plist Modification	DLL Side-Loading	Private Keys	System Network Configuration Discovery	Windows Remote Management		Remote File Copy		
	Regsvr32	File System Permissions Weakness	Port Monitors	Execution Guardrails	Securityd Memory	System Network Connections Discovery			Standard Application Layer Protocol		
	Rundll32	Hidden Files and Directories	Process Injection	Exploitation for Defense Evasion	Two-Factor Authentication Interception	System Owner/User Discovery			Standard Cryptographic Protocol		
	Scheduled Task	Hooking	Scheduled Task	Extra Window Memory Injection		System Service Discovery			Standard Non-Application Layer Protocol		
	Scripting	Hypervisor	Service Registry Permissions Weakness	File Deletion					Uncommonly Used Port		
	Service Execution	Image File Execution Options Injection	Setuid and Setgid	File System Logical Offsets					Web Service		
	Signed Binary Proxy Execution	Kernel Modules and Extensions	SID-History Injection	Hidden Files and Directories							
	Signed Script Proxy Execution	Launch Agent	Startup Items	Image File Execution Options Injection							
	Source	Launch Daemon	Sudo	Indicator Blocking							
	Space after Filename	Launchctl	Sudo Caching	Indicator Removal from Tools							
	Third-party Software	LC_LOAD_DYLIB Addition	Valid Accounts	Indicator Removal on Host							
	Trap	Local Job Scheduling	Web Shell	Install Root Certificate							
	Trusted Developer Utilities	Login Item		InstallUtil							
	User Execution	Logon Scripts		Launchctl							
	Windows Management Instrumentation	LSASS Driver		Masquerading							
	Windows Remote Management	Modify Existing Service		Modify Registry							
	XSL Script Processing	Netsh Helper DLL		Mshta							
		New Service		Network Share Connection Removal							
		Office Application Startup		NTFS File Attributes							
		Path Interception		Obfuscated Files or Information							
		Plist Modification		Plist Modification							
		Port Knocking		Port Knocking							
		Port Monitors		Process Doppelgänging							
		Rc.common		Process Hollowing							
		Re-opened Applications		Process Injection							
		Redundant Access		Redundant Access							
		Registry Run Keys / Startup Folder		Regsvcs/Regasm							
		Scheduled Task									
		Screensaver									
		Security Support Provider									
		Service Registry Permissions Weakness									
		Setuid and Setgid									
		Shortcut Modification									
		SIP and Trust Provider Hijacking									
		Startup Items									

Procedures: Specific technique implementations

Name	Description
APT29	APT29 used sticky-keys to obtain unauthenticated, privileged console access.[4][5]
APT3	APT3 replaces the Sticky Keys binary `C:\Windows\System32\sethc.exe` for persistence.[6]
Axiom	Axiom actors have been known to use the Sticky Keys replacement within RDP sessions to obtain persistence.[7]
Deep Panda	Deep Panda has used the sticky-keys technique to bypass the RDP login screen on remote systems during intrusions.[8]
Empire	Empire can leverage WMI debugging to remotely replace binaries like sethc.exe, Utilman.exe, and Magnify.exe with cmd.exe.[9]

Red Team Øvelse

1. Innledende rekognosering

Den første fasen i en Red Team-øvelse fokuserer på å samle så mye informasjon som mulig om målet. Dette bygger også på trusseletteretning.

2. Kartlegging av ekstern angrepsflate

Red team vil utføre aktiv skanning på en forsiktig måte for å kartlegge og verifisere angrepsflaten samtidig som de unngår å bli oppdaget av blue team. Red team vil videre analysere informasjonen som er samlet inn om infrastrukturen, fasilitetene og ansatte.

3. Initiell kompromittering

Leveranse av skreddersydde angrep (payloads) og utnyttelse for å nå målene eller flaggene gjennom aktiviteter som sosial manipulasjon, utnyttelse av tekniske sårbarheter, fysisk testing som involverer planting av hardware-trojanere for nettverkspersistens, osv.

4. Etabler fotfeste

Persistering og unngå forsvarsmekanismer for å opprettholde tilgang internt.

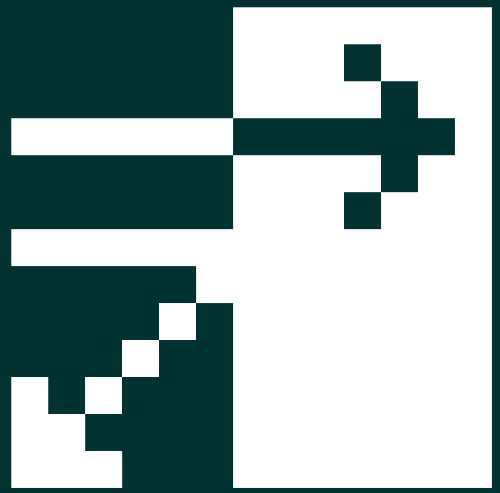
5. Interne aktiviter

- Intern rekognosering
- Eskalering av privilegier
- Lateral bevegelse
- Opprethold persistering

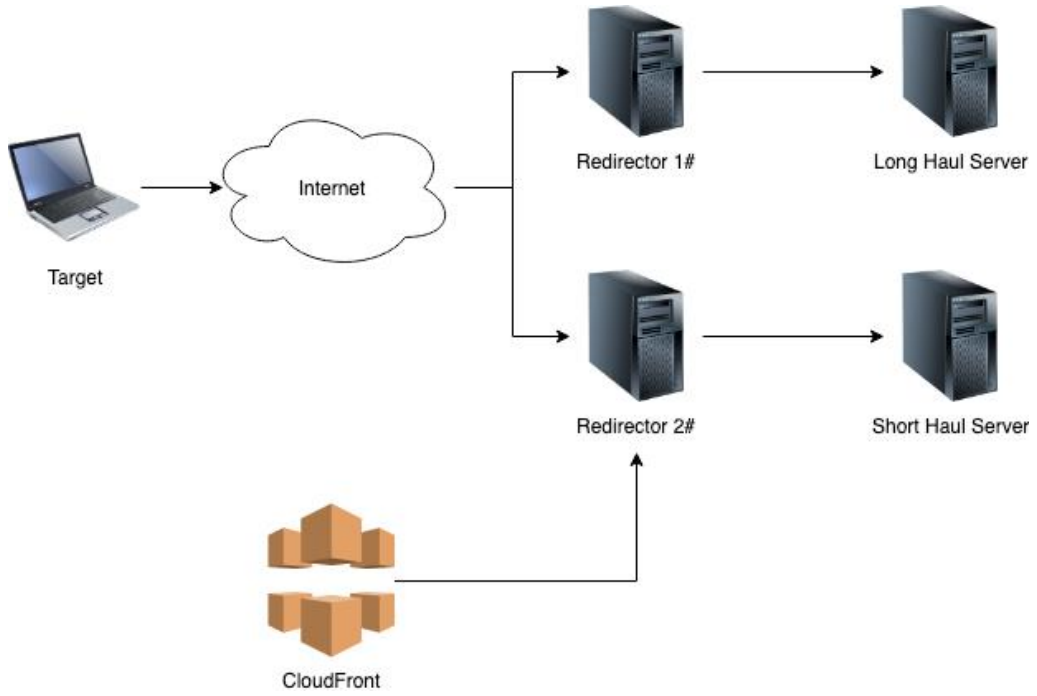
6. Fullføre test

Mulig eksfiltrering av data og oppnå de målene som ble avtalt under planleggings- og scopingmøter

Eksempel Angrep



Command and Control (C2) Infrastruktur



- Command and Control infrastruktur for fjernstyring og persistering på klienter og servere
- Kompromittert klient/server kontakter spesifikt domene
- Benytter gjerne domener merket som «helse» eller «finans» for å omgå TLS inspeksjon
- «Domain fronting» med bruk av f.eks Cloudflare eller Azure

OSINT

Informasjon direkte knyttet til firma

- Hvor mange ansatte har bedriften?
- Hvor store kundeportefølje har bedriften?
- Hvilke samarbeidspartnere har bedriften?
- Hvilke leverandører har bedriften?
- Er bedriften en del av et større konsern eller over/under selskap?

Informasjon knyttet til ansatte og personell

- Hvilke ansatte fyller sentrale roller i bedriften?
- Navn, epost-adresser, telefon og adresse
- Finnes tidligere lekkede passord og brukernavn?
- Sosiale medier - Linkedin

Teknisk informasjon knyttet til firma

- Hva slags infrastruktur er tilgjengelig fra internett?
- Hvilken driftsleverandør benytter bedriften?
- Sier tidligere innsamlet informasjon noe om modenhet / IT-sikkerhet?

Faser

Rekognosering

Open Source Intelligence

- Samlet sammen brukerliste fra åpne kilder
- Veldig mange brukere og passord ble funnet i åpne passordlekkasjer
- For å verifisere brukernavn/passord, ble et internett-eksponert system brukt for passordspraying.

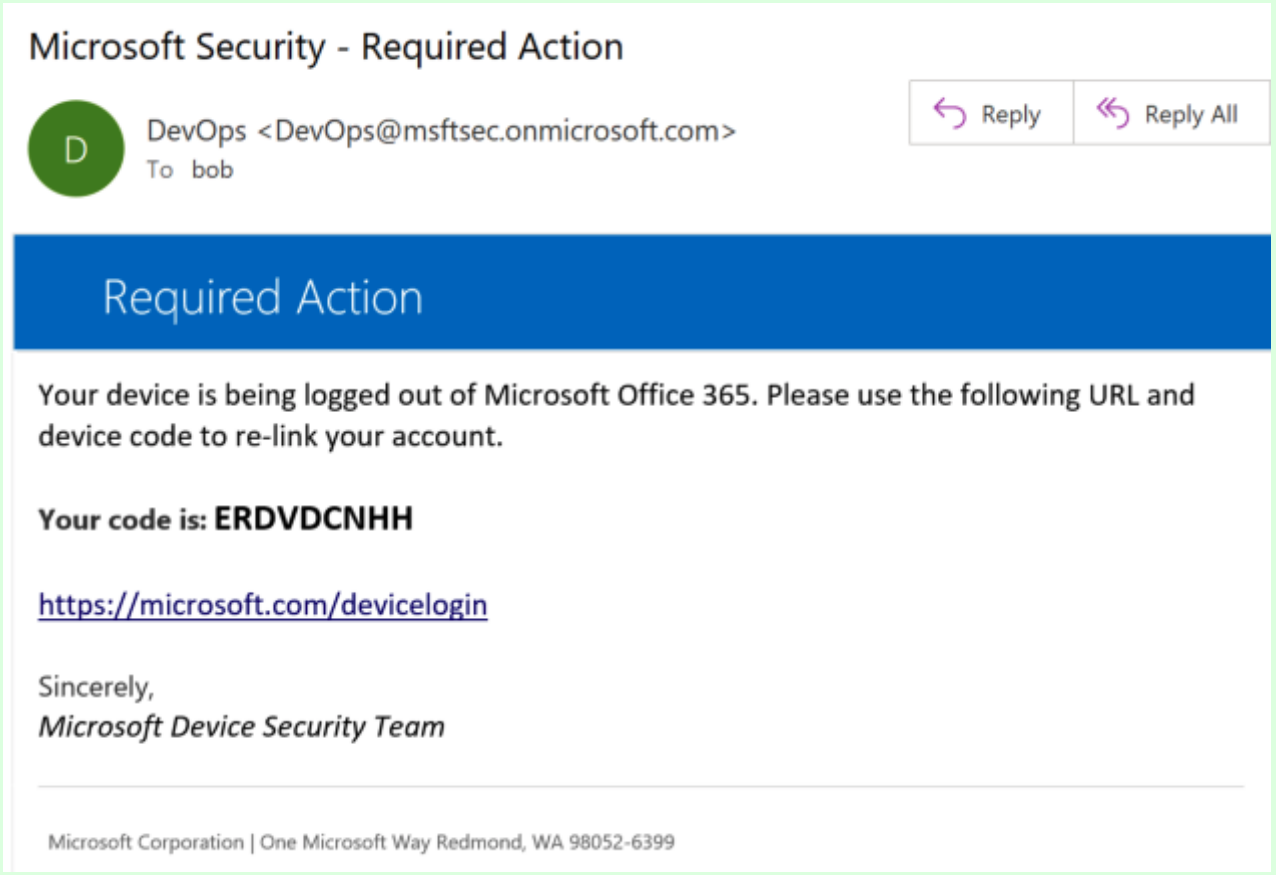
Faser

Rekognosering

↓

Sosial Manipulering

Device Code Phish



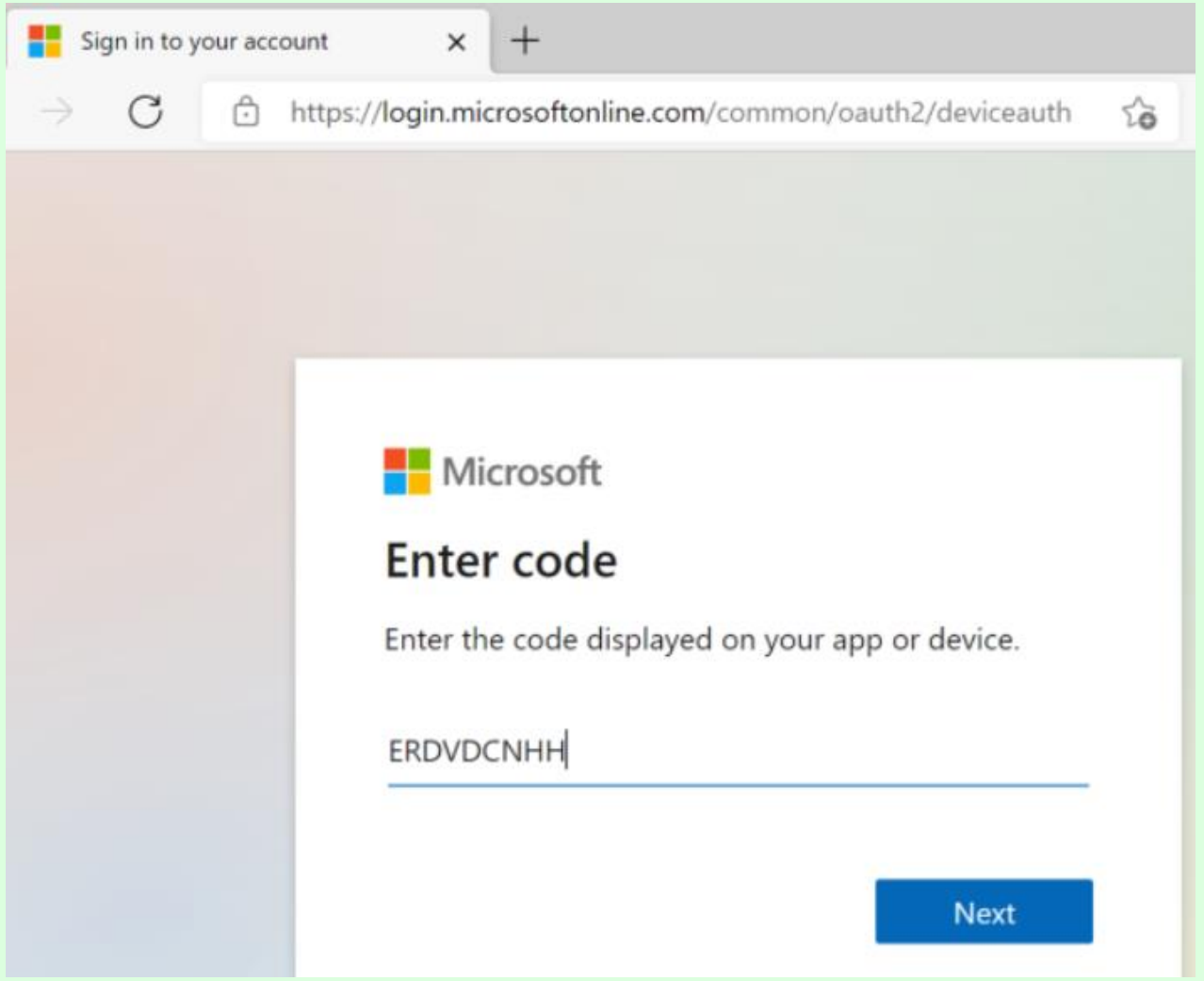
Faser

Rekognosering

↓

Sosial Manipulering

Device Code Phish



Faser

Rekognosering

↓

Sosial Manipulering

Device Code Phish

[dcphy][]->: token:request

[+] Device code requested: D7C43ZQNQ

[-] Waiting for user authentication on https://microsoft.com/devicelogin with code: D7C43ZQNQ

[-] This code will expire on 2022-09-14 11:07:12.627128.

[+] Success! User authenticated!

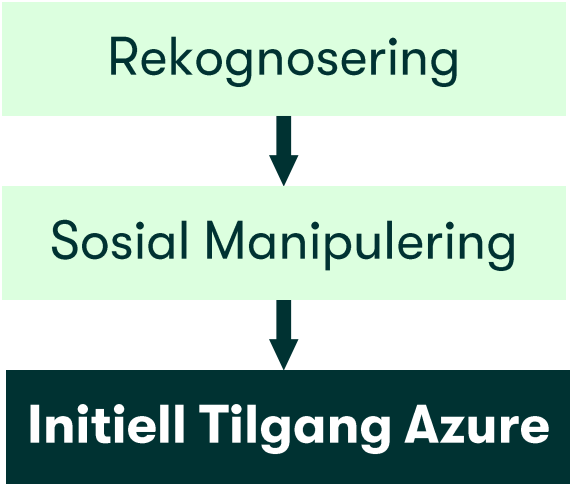
[dcphy][]->: token:status

[-] Token is active:

Access token:

eyJ0eXAiOiJKV1QiLCJub25jZSI6Imd6Q2xxZzNpVmlZRTMyRnZGelBGNWF4amFFallLeS1sZlJLeHZDLTRlcEUiL
CJhbGciOiJSUzI1NiIsIng1dCI6IjJaUXBKM1VwYmpBWVhZR2FYRUpsOGxWMFRPSSIsImtpZCI6IjJaUXBKM1V
wYmpBWVhZR2FYRUpsOGxWMFRPSSJ9.eyJhdWQiOiJodHRwczovL2dyYXBoLm1pY3Jvc29mdC5jb20iLCJp
c3MiOiJodHRwczovL3N0cy53aW5kb3dzLm5ldC9kYzJjZmE3Ni1iZTY0LTQxZDMtYjljMy1hZTk5NzVhZDkzNj
QvIiwiaWF0IjoxNjYzMTQ1MzY0LCJuYmYiOjE2NjMxNDUzNjQsImV4cCI6MTY2MzE1MDk0MCwiYWNjdCI6
MCwiYWNyIjoiMSIsImFpbyI6IkFWUUFxLzhUQUFBQXh6MGo1Wm4yU0w1cVVxa2dodTkvT0kzTE96eEMw

Faser

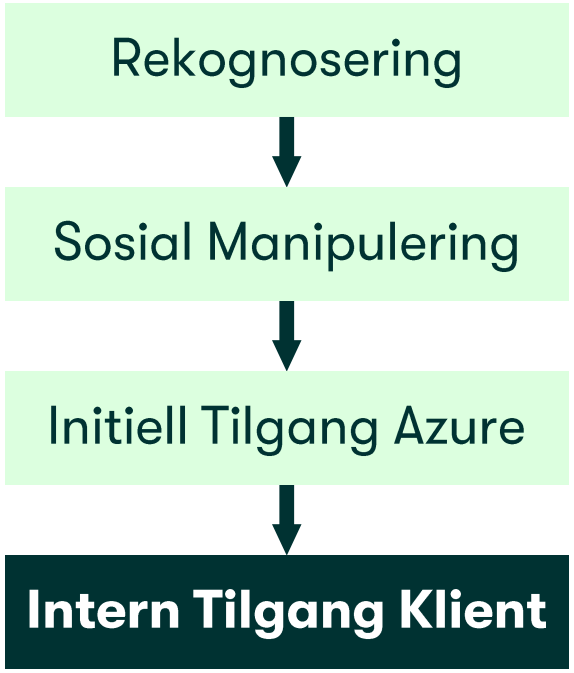


Rekognosering og skanning i Azure med bruker

Scanning av Sharepoint for sensitiv data

- Liste brukere og informasjon
- Tilgang
  - E-mail
  - Teams
  - Sharepoint
  - Onedrive

Faser



Intern rekognosering

Kjørende agent på klient

Intern rekognosering i lokalt nettverk – port scanning

- SMB – Fileshares / Fildeling
- RDP – Fjerninnlogging
- WMI – Fjernadministrering og monitorering

Begrenset nettverkstilgang til servere

Faser

Rekognosering

↓

Sosial Manipulering

↓

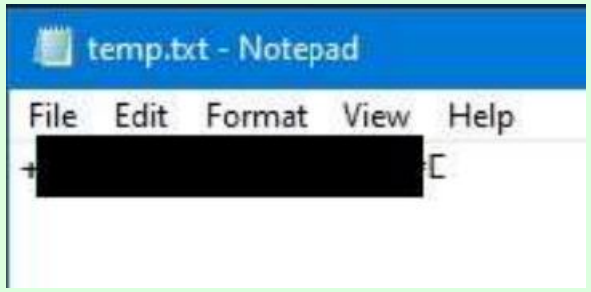
Initiell Tilgang Azure

↓

Intern Tilgang Klient

Eskalering av privilegier til lokal server administrator

Tekstfil `temp.txt` lesbar av alle domenebrukere funnet på åpen fileshare



Faser

Rekognosering

↓

Sosial Manipulering

↓

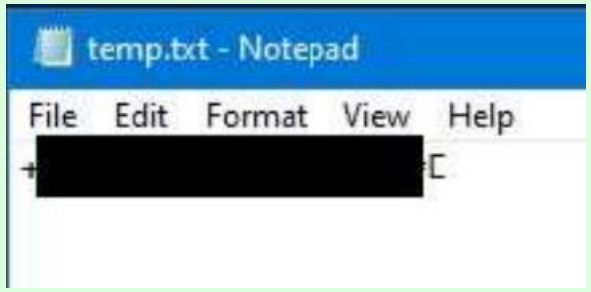
Initiell Tilgang Azure

↓

Intern Tilgang Klient

Eskalering av privilegier til lokal server administrator

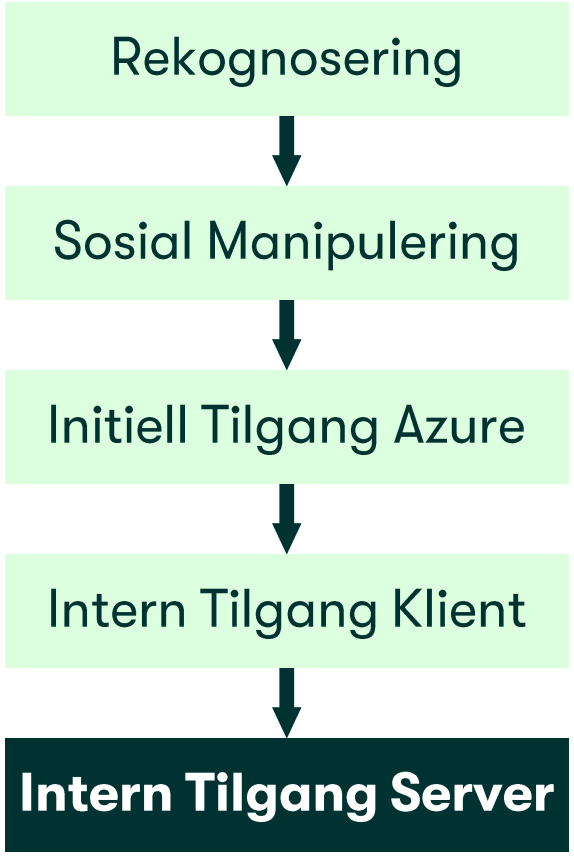
Tekstfil `temp.txt` lesbar av alle domenebrukere funnet på åpen fileshare



Passordspraying av alle brukere i Active Directory med denne strengen

- Streng gyldig passord for en tjenestekonto

Faser

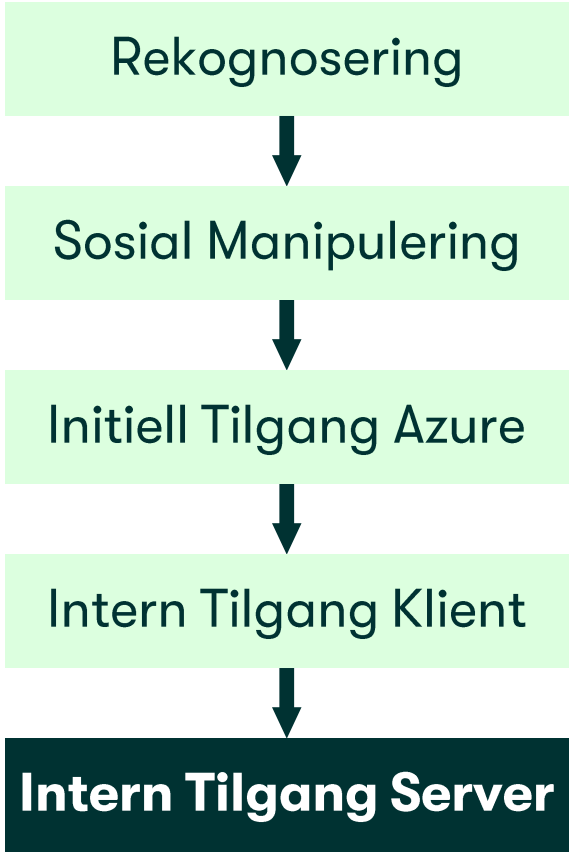


Lateral bevegelse og eksalering av privilegier server administrator

Startet agent på server

- Fra server tilgang til flere nettverkssoner internt

Faser



Lateral bevegelse og eksalering av privilegier server administrator

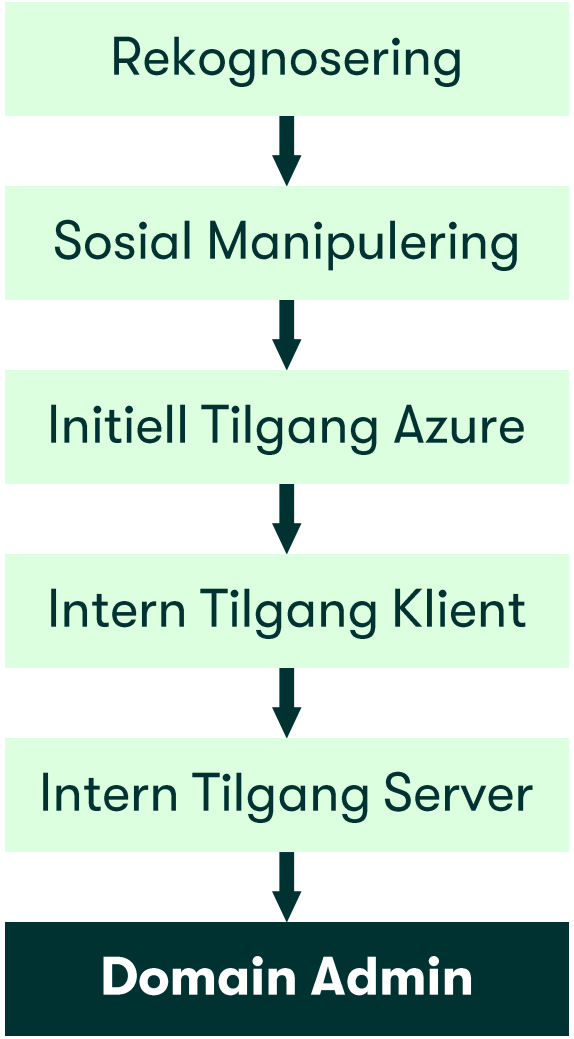
Startet agent på server

- Fra server tilgang til flere nettverkssoner internt

Administrator session hijacking

- En server administrator logget inn på to av servere vi har tilgang til

Faser



Lateral bevegelse og eksalering av privilegier domene administrator

Domene administrator logget inn på server vi nå har tilgang til

- Bypass antivirus og dumping av minne for brukernavn/passord
    - procdump.exe –ma lsass.exe outfile.dmp
- Lokal verktøy for å hente brukernavn/passord fra minne

Faser

Rekognosering

↓

Sosial Manipulering

↓

Initiell Tilgang Azure

↓

Intern Tilgang Klient

↓

Intern Tilgang Server

↓

Domaim Admin

Lateral bevegelse og eksalering av privilegier domene administrator

Domene administrator logget inn på server vi nå har tilgang til

- Bypass antivirus og dumping av minne for brukernavn/passord
  - procdump.exe –ma lsass.exe outfile.dmp
- Lokal verktøy for å hente brukernavn/passord fra minne

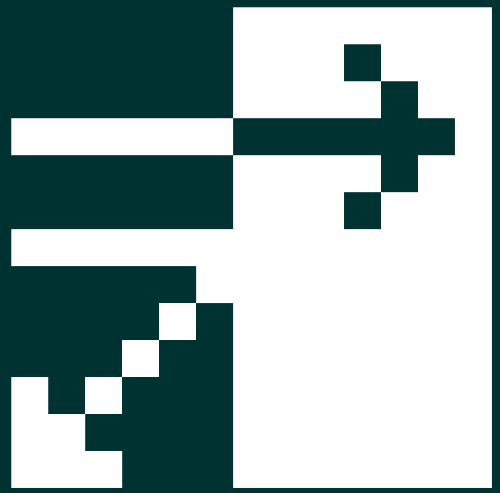
```
mimikatz # sekurlsa::minidump lsass.DMP
Switch to MINIDUMP : 'lsass.DMP'

mimikatz # log lsass.txt
Using 'lsass.txt' for logfile : OK

mimikatz # sekurlsa::logonPasswords
Opening : 'lsass.DMP' file for minidump...

Authentication Id : 0 ; 204847 (00000000:0003202f)
Session           : Interactive from 1
User Name         : test
Domain            : DESKTOP-3334N46
Logon Server      : DESKTOP-3334N46
Logon Time        : 7/22/2021 9:49:47 AM
SID               : S-1-5-21-114072906-1243854157-2166989921-1000
        msv :
         [00000003] Primary
         * Username : test
         * Domain   : DESKTOP-3334N46
         * NTLM     : 31d6cfe0d16ae931b73c59d7e0c089c0
```

Fysisk Testing



Fysiske Test Scenarioer

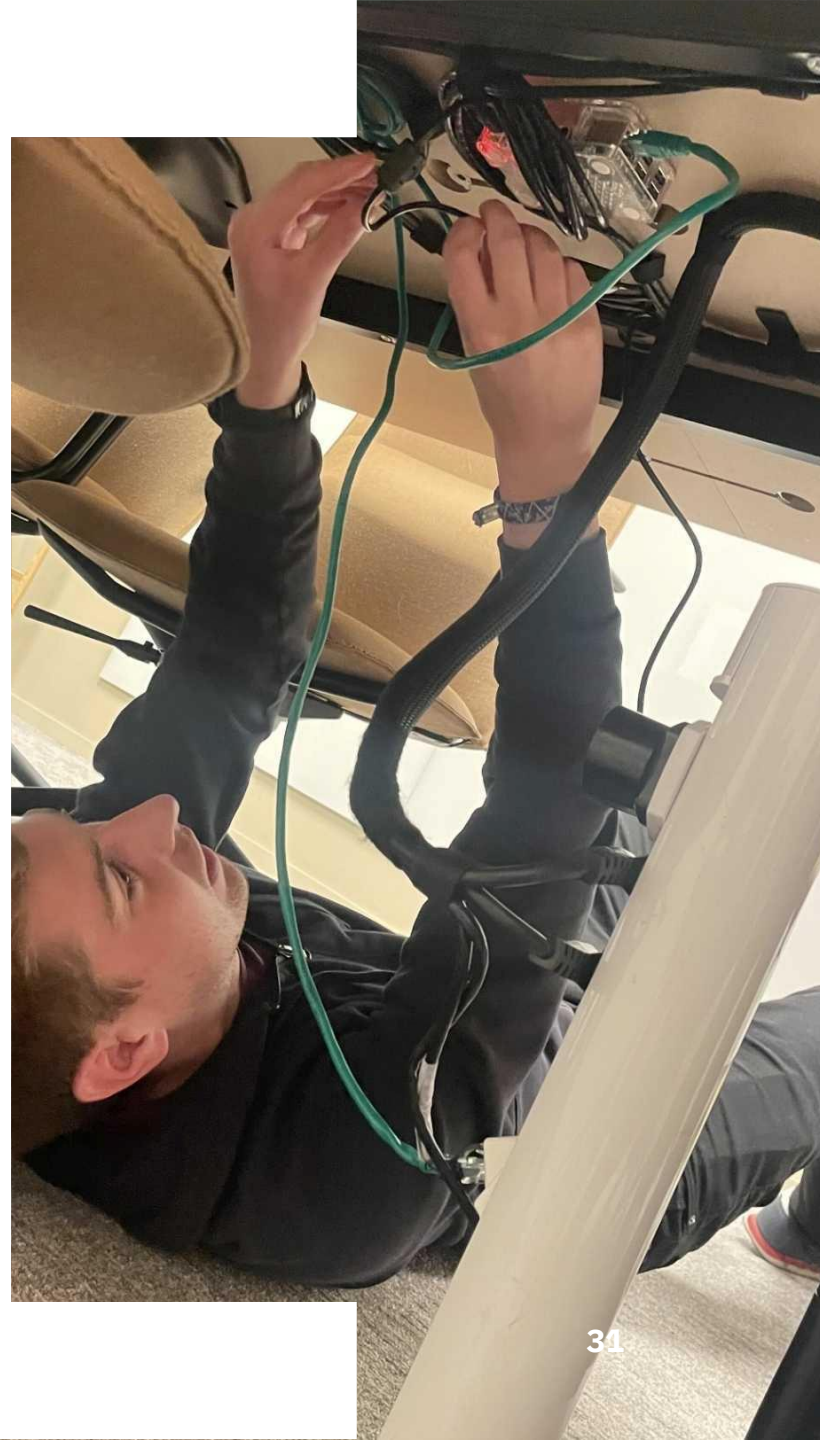
Undersøk muligheter som uvedkommende har til å oppnå fysisk tilgang til lokale nettverk på lokale kontor og hovedkontor

Rekognosering

- Undersøkelser i åpne kilder
- Befaring på stedet for å undersøke mulige innganger og utganger og vurdere ulike fysiske sikkerhetstiltak

Angrepsforsøk

- Kopiering av adgangskort
  - Sluppet inn av rengjøringspersonell på en lørdag
- Tailgating vanlig arbeidsdag
- Installerte raspberry internt og omgikk NAC



Oppsummering

Tiltak

Sikkerhet må bygges inn tidlig i planlegging, utvikling og testing.

Forebygg tidlig ved å:

- Støtte trusselmodellering i designfasen
- Inkludere sikkerhetstest-caser og penetrasjonstesting i testplaner
- Sørge for at sårbarheter følges opp og prioriteres

Funn fra penetrasjonstester er starten på forbedring – Sikkerhet er en kontinuerlig prosess

Spørsmål og avslutning

- Penetrasjonstesting og red teaming er viktige verktøy – ikke bare for IT, men hele organisasjonen.
- Målet er ikke bare å finne feil – men å bygge tillit til løsningene våre.
- Sikkerhet er et felles ansvar

Thank you!