Data Exfiltration Detection

Data Exfiltration is the unauthorized transfer of sensitive data.

→ It can be deliberate (insider) or via malware/compromised credentials

Reasons:

- Financial gain
- Espionage
- Ransomware & Extortion
- Disruption & Sabotage
- Persistence & Reconnaissance

Examples

APT 29 (cozy bear) → HTTPS over legitimate domains → Encrypted HTTPS channels were used to exfiltrate data from government networks

FIN7 → HTTP Post to C2 servers → Embedded data in HTTP POST requests to evade detection

LUNAR SPIDER → Encrypted C2 channel → Maintained a 2 month intrusion using encrypted channels and staged exfiltration

Darkside Ransomware → Dual exortion: encryption + exfil → Stole data before encrypting systems, then threatened public leaks

APT 10 (Cloud Hopper) → Cloud to Cloud Transfer → Exfiltrated data from managed service providers using cloud APIs

Techniques And Indicators

Techiques	Examples	Indicator of Attack and where to look
Network-based	HTTP/HTTPs uploads to S3/Azure/webmail, FTP/SFTP/SCP, DNS tunnelling, ICMP/covert protocol, custom TCP/UDP	Proxy/web gateway logs (large POSTs, uploads to cloud endpoint), firewall/NGFW flows (high bytes to single IP/ASN) netflow (spikes/outbound flows) DNS logs (long hostnames TXT queries)
Host-based	Powershell Invoke-WebRequest, rclone awscli, curl/wget - archive creation (zip/rar) -use of removable USBs - ADS/hidden streams	Sysmon/EDR (Process Create Network Connect, File Creat Windows Security (4633/4656 object access) auditd/shell history on Linux and removable-media events.
Cloud exfiltration	- S3 PutObject/multipart upload - Azure Blob upload - Google Cloud Storage objects - Insert. Drive/sharePoint ~~extend~~ external sharing	CloudTrail/Azure Activity GCP Audit, cloud storage access logs, unusual service-account or IP activity.

Covert & encoding	- DNS tunnelling, Base64 or chunked encoding steganography into images - splitting files into many small requests	- DNS logs - proxy logs with many small POSTs - correlation of intermittent uploads + suspicious process activity.
Insider & collaboration tools	Slack/Teams/Dropbox Google Drive/Box uploads or sharing to external users;	- Audits logs, and mail logs
General IoAs & triage signals	- large outbound volume to external IPs/domains - unknown destination domains - suspicious processes/command lines - many file read events followed by an outbound connection - multipart/streamed uploads	- Correlate: Proxy, Firewall, Netflow, DNS, Sysmon/EDR, mail server logs.

DNS tunneling

smuggle Bytes encoded inside DNS queries/responses to firewalls and webproxies don't notice.

DNS queries look like normal requests

DNS data exfiltration Indicators:

- Many DNS querries are sent to single external domain
- Long ~~dor~~ subdomain
- High entropy or Base 32/ Base 64 patterns
- (TXT, Null) record types
- Unusual response Behaviour
- Queries at regular intervals

Questions (using Wireshark)

1) What is the suspicious domain receiving the DNS traffic?

tunnelcorp.net

2) How many suspicious traffic/logs related to dns tunneling were observed?

dns && dns.qry.name contains "tunnelcorp.net"

3) Which local IP sent the maximum number of suspicious requests?

192.168.1.103 (72 packet)

FTP Data exfiltration

File Transfer Protocol is one of oldest protocols for transferring files over TCP/IP network

Attackers use:
- → legitimate FTP server
- → compromised credentials
- → non-standard ports (Blend with other traffic)

Indicators:
- → User and Pass commands
- → STOR and RETR commands
- → large data connections to ~~unusual~~ unusual IPs
- → Data channel openings on ephemeral ports (PASV) paired with large payloads.

Questions (used Wireshark)

1) How many connections were observed from the guest account?

ftp && ftp.request.arg == "guest"

2) Apply the filter, what is the name of the customer related file exfiltrated from the root account?

customer_data.xlsx

3) which internal IP was found to be sending the largest payload to an external IP

192.168.1.105 (length 127)

4) What is the flag hidden inside the ftp stream transferring the CSV file to the suspicious IP?

found

Data Exfil via HTTP

- Blends with normal web traffic
- HTTP is very common

How use HTTP for data exfil

→ POST upload to external servers
→ GET request with encoded data
→ Use of common services / CDN
→ Custom headers
→ Chunked transfer / multipart
→ HTTPS / TLS tunneling
→ Staging via cloud services

Indicators:

→ Unusual large HTTP Post request
→ HTTP requests to domains with low reputation
→ Frequent small requests to the same host
→ Chunked or multipart transfer

Questions

1) Which internal compromised host was used to exfiltrate this sensitive data?

http.request.method == "POST" and frame.len > 750

2) What's the flag hidden inside the exfiltrated data?

Follow → HTTP Stream

Data exfiltration via ICMP

ICMP used for diagnostics and control (ping, TTL exceeded)

How use ICMP for exfil

Common Techniques:

→ ICMP echo (type 8)/reply (type 0) tunnelling: place encoded inside ICMP
→ Custom ICMP types/codes
→ Fragmentation and reassembly: split across multiple packets
→ Encryption/obfuscation

Indicators:

→ Persistent ICMP sessions
→ large ICMP payloads or frequent ICMP
→ Contain high-entropy data
→ Bursts of ICMP

Indicators in Wireshark:

→ ICMP packet volume
→ Large frame.len or icmp.payload
→ ICMP type
→ Regular timing
→ Fragments with reassembly

Question

1) what is the flag found in the exfiltrated data through ICMP?

```
icmp.type == 8 && frame.len > 32
```