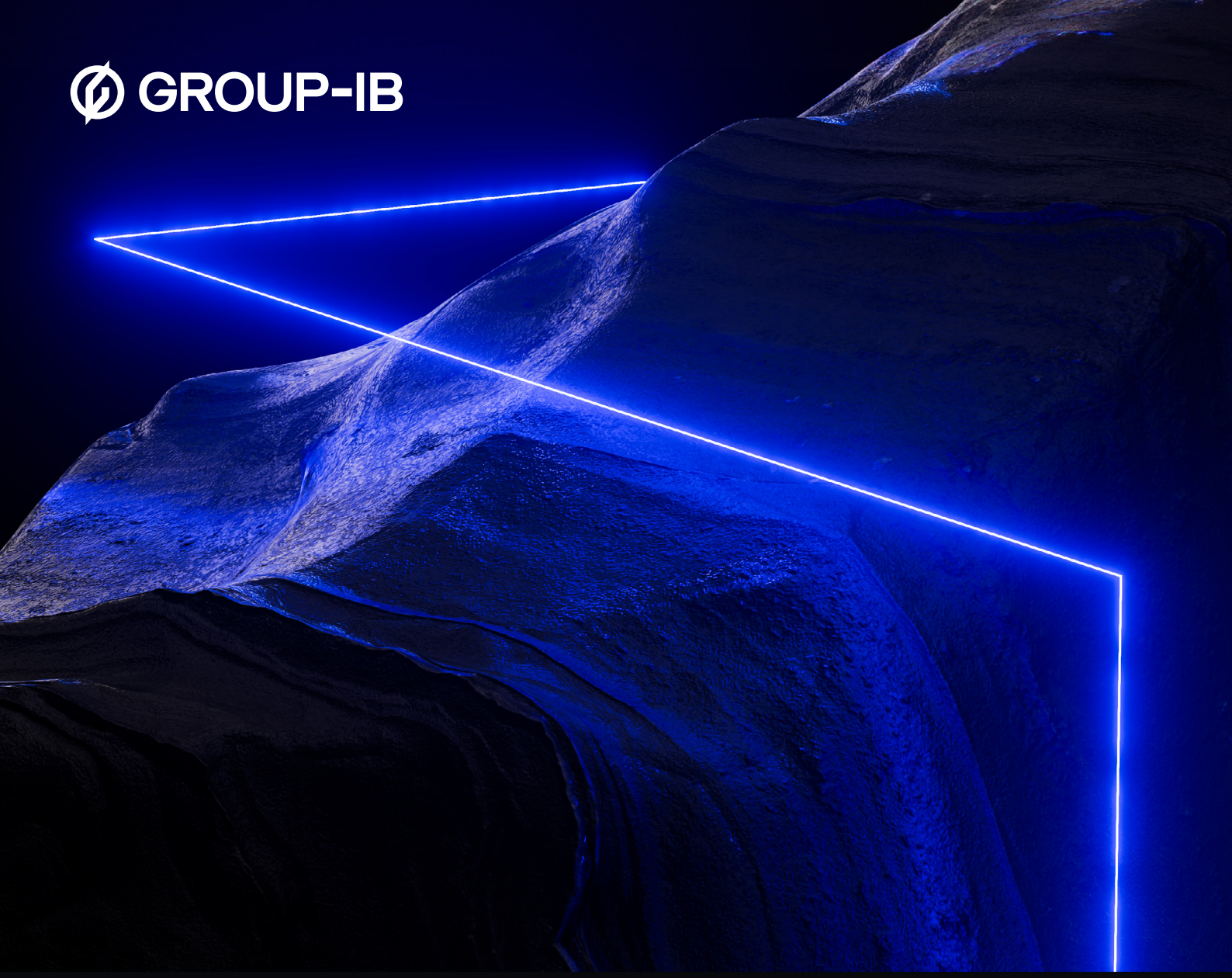
GROUP-IB



HANDBOOK

RECONNAISSANCE HANDBOOK

Map and mitigate intrusion pathways into your network

GROUP-IB.COM

TABLE OF CONTENTS

1. INTRODUCTION: WHAT IS RECONNAISSANCE?

Not all cyberattacks are created equal. They might involve different underlying motives, TTPs used, or targets. Before they initiate a cyberattack, adversaries usually carry out an in-depth analysis and plan their strategy.

An attack can be broken down into several broad phases, commonly referred to as the "kill chain" (which can be pivoted depending on the type of attack). Various security frameworks (such as the Cyber Kill Chain* and MITRE ATT&CK) recognize reconnaissance as the fundamental stage in any type of attack.

Reconnaissance is used for planning cyber intrusions (by adversaries) and for strengthening security (by cyber defenders). Adversaries leverage reconnaissance to gather information about target systems, understand their organizational infrastructure and potential attack vectors, and collect additional details that will help make the attack successful and assess its impact.

Reconnaissance is a way for adversaries to observe their target environment, orient themselves, and make informed decisions about any subsequent steps in their attack kill-chain. By exploring the extent of their control and identifying entry points, adversaries gain insights into how these factors can help further their goals.

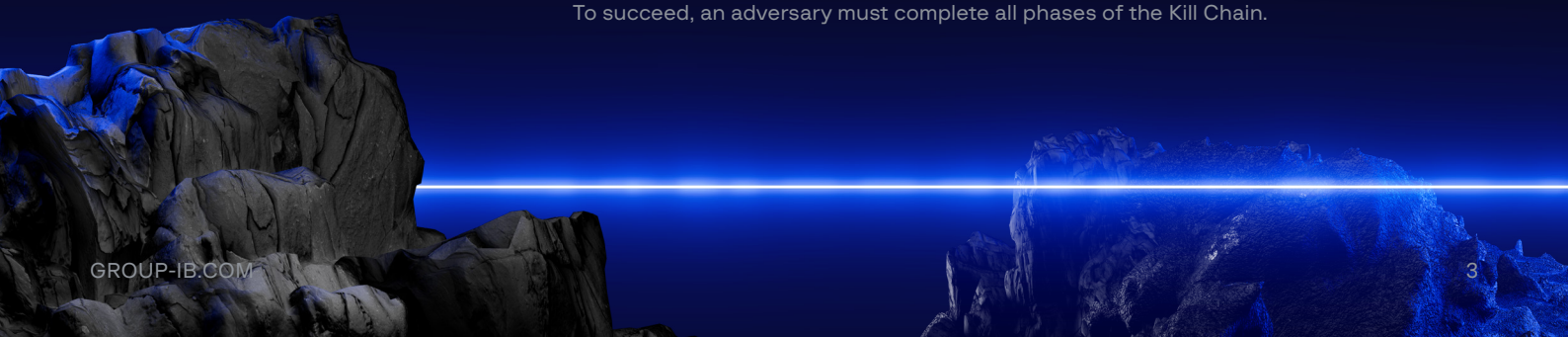
When successful, reconnaissance helps execute a wide range of cyberattacks, ranging from ransomware and data exfiltration to additional reconnaissance, leaks, lateral movement, and more. It essentially opens the door to a range of malicious activities.

On the flip side, cybercriminals aren't the only ones who can use reconnaissance as a helpful tool — cyber defenders can use it to their advantage as well. Reconnaissance can be a powerful defense tactic when used before active operations in penetration testing, red teaming exercises, and security assessments. Obtaining detailed information about a target system, coupled with gaining an understanding of exploitable vulnerabilities, means that defenders can initiate corrective actions based on strategic security assessments. Such a proactive approach significantly enhances an organization's resilience against potential cyber threats.

The dual nature of reconnaissance as both an offensive and defensive strategy highlights its importance in the ever-evolving landscape of cybersecurity.

---

*Cyber Kill Chain is a framework for the cybersecurity industry that defines the sequential steps used by adversaries or malicious actors in cyberspace to plan and execute attacks. To succeed, an adversary must complete all phases of the Kill Chain.



2.

WHY IS RECONNAISSANCE IMPORTANT?

In cybersecurity, the reconnaissance stage can be compared to a scene from a spy movie where the protagonist gathers information about the enemy's base before setting off on a mission. This might involve studying satellite images, considering possible entrances and exits, and conducting on-site surveillance. Such preparation steps are essential for shaping the attack and refining its approach before the final stage is executed. In the context of cybersecurity, penetration testers consider the reconnaissance stage just as important and recognize the need for gathering thorough and accurate information before going on the offensive.

Much like in the high-stakes world of vigilantes and their missions, pen-testers understand the value of getting things right the first time — there often isn't room for second chances.

Assessing the security of information systems requires a comprehensive understanding of the target, which makes reconnaissance critical in penetration testing and red teaming exercises. Many reconnaissance methods exist and a single document may not be enough to cover all of them. In this guide, Group-IB's pen-testers shed light on a few basic yet integral approaches used during the reconnaissance phase.

Combining manual techniques with automation tools enables attackers and security professionals to identify the attack surface. Security specialists and network administrators may not always be fully familiar with all of their organization's assets and weaknesses. Penetration testing often brings forth unused and vulnerable assets, as well as shadow IT (any software, hardware, or IT resource used on a network without the IT department's approval and/or knowledge), which can pose significant risks. The consequences could be highly damaging and additional systems within the organization could be affected.

Reconnaissance is therefore an important skill not only for penetration testing specialists but also for system administrators and security professionals.

To manage and monitor the attack surface, organizations can leverage solutions that fall under what's called Attack Surface Management (ASM). While ASM may be invaluable for internal reconnaissance, it is not a substitute for manual assessments. The three-way synergy of (1) expert-driven efforts, (2) ASM tools, and (3) consistent monitoring of asset inventories should be used as a frontline approach to improving organizational security. The goal is to be up to date with the threat landscape and prioritize proactive measures.

3. ACTIVE VS. PASSIVE RECONNAISSANCE

When performing penetration testing, cyber reconnaissance is the key step, and there are two broad approaches to gathering information and understanding network and system topology. There are essentially two ways to obtain information about a target: interact with the target itself or ask a third party about the target. Active and passive reconnaissance reflect these two ways of gathering information.

While this makes for the basic definition of both processes, there's more to both active and passive reconnaissance.

Approach	Active Reconnaissance	Passive Reconnaissance
Activity	Direct interaction with the target, often through automated tools or actions	Information is collected without direct interaction with the target
Examples	• Rapid, automated requests • Testing for specific vulnerabilities • Interacting with the target's systems	• Asking a third party for information • Gathering data without direct interaction
Intent and impact	The classification depends on technical differences and on the intent and potential impact of actions	Considered based on the "reasonable person" standard by weighing intent and impact
Automated tools	Use of automated tools that may degrade service availability	Not inherently involving automated tools that impact service availability
Testing admin panel	Testing for the existence of an exposed admin panel (considered active due to potential security concerns)	The exposed admin panel displayed openly
Nuance	• Intent and potential impact play a crucial role in classification • Automated tools can have varying degrees of impact)	• Third-party interactions may involve various scenarios • Intent and impact are critical considerations
Clarity of line	The line between active and passive reconnaissance is nuanced, considering technical differences, intent, and impact	The activities under passive recon are not distinctly clear-cut, and are performed once they're considered legally and ethically safe.
Third-party interaction	Not directly related to asking third parties for information about the target	Asking third parties for information, with consideration given to legality and source reliability
High-level heuristics	• More interaction with the target • Automatead • More potential negative impact • Larger volume • Go as fast as it is physically possible • "Respect" only technical barriers • More intrusive	• Little interaction with the target • Manual • Little potential negative impact • Smaller volume • When in doubt, apply the rate limit • Act according to the intent specified by the client • Less intrusive

Figure 1.
Performing reconnaissance (automated review)

```
┌──(kali㉿kali)-[~]
└─$ amass enum -d group-ib.com
group-ib.com (FQDN) --> ns_record --> tom.ns.cloudflare.com (FQDN)
group-ib.com (FQDN) --> ns_record --> dora.ns.cloudflare.com (FQDN)
group-ib.com (FQDN) --> node --> fhp-aws-antibot.group-ib.com (FQDN)
group-ib.com (FQDN) --> node --> team.group-ib.com (FQDN)
group-ib.com (FQDN) --> node --> sbbe.group-ib.com (FQDN)
group-ib.com (FQDN) --> node --> sgbe.group-ib.com (FQDN)
group-ib.com (FQDN) --> node --> partneracademy.group-ib.com (FQDN)
fhp-aws-antibot.group-ib.com (FQDN) --> cname_record --> k8s-istiosys-istioing-bf0a58c
.com (FQDN)
team.group-ib.com (FQDN) --> cname_record --> ghs.googlehosted.com (FQDN)
sbbe.group-ib.com (FQDN) --> cname_record --> sgbe.group-ib.com (FQDN)
partneracademy.group-ib.com (FQDN) --> cname_record --> lms1eskj79162fo6bwlaxks5izi0s9
soc.group-ib.com (FQDN) --> a_record --> 31.184.221.98 (IPAddress)
people.group-ib.com (FQDN) --> cname_record --> ghs.googlehosted.com (FQDN)
partners.group-ib.com (FQDN) --> cname_record --> groupibgipp.prod.impartner.live (FQD
group-ib.com (FQDN) --> mx_record --> mx3.atmosphere.group-ib.com (FQDN)
group-ib.com (FQDN) --> mx_record --> mx1.atmosphere.group-ib.com (FQDN)
group-ib.com (FQDN) --> mx_record --> mx2.atmosphere.group-ib.com (FQDN)
group-ib.com (FQDN) --> mx_record --> mx1.atmosphere.sg.group-ib.com (FQDN)
group-ib.com (FQDN) --> mx_record --> mx2.atmosphere.sg.group-ib.com (FQDN)
```

NB: Although these tools yield good results, they do often fall short and may not use some of the techniques outlined above. Manual review therefore remains essential to validate or dismiss any findings.

To both simplify and enhance the reconnaissance process, Group-IB experts use their proprietary internal data lake, one of the largest in the industry. This resource provides unique insights that can be used in many ways, which is useful not only in reconnaissance involving Attack Surface Management but also in supporting various security initiatives such as hi-tech crime investigations, threat intelligence, and fraud protection.

The data lake serves as a source of information, which can be tailored in a way that helps to understand an organization's digital landscape as well as identify potential vulnerabilities and emerging threats. It offers a comprehensive view of an organization's security posture.

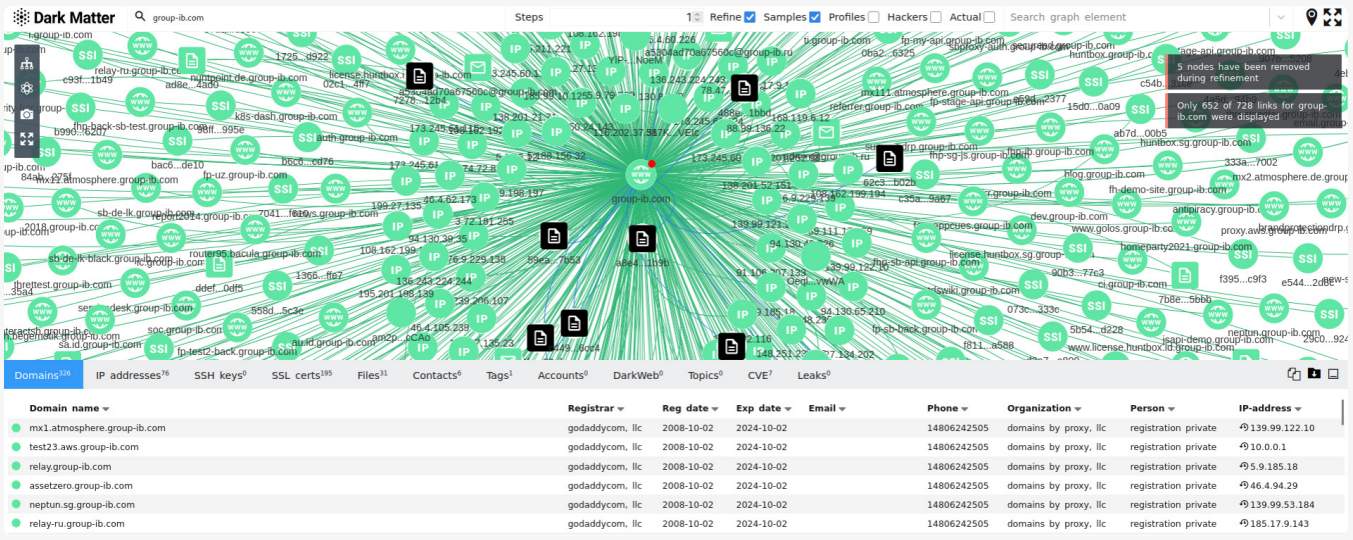


Figure 2.
Screen showing the data points captured by Group-IB's Threat Intelligence

4. SUGGESTED COURSE OF ACTION FOR COMMON RECONNAISSANCE CASES

Automated DNS tools in semi-passive subdomain gathering

The results of passive subdomain discovery tools often contain subdomains that do not resolve anything, either due to mistakes in the data source or because the subdomains were simply discontinued. We use automated tools to filter out those that are still valid. This is considered semi-passive reconnaissance if proper precautions are taken. First, it is recommended to use a list of trustworthy public DNS resolvers to spread out a load of DNS queries. Public DNS servers have better cache locality so they don't need to contact the target's authoritative server as much, and when they do it is from various public DNS servers instead of the stub resolver on your network. This also prevents you from accidentally DOSing your local stub DNS resolver (yes, this has happened, and more than once). Second, make sure that the candidate subdomains list already mostly consists of valid subdomains. Other than volume and rate, this is the crucial difference between brute-forcing subdomains (very active) and verifying passive subdomain discovery. You cannot append a huge list of common subdomains, run it through automated tools to filter out the good ones, and call it passive reconnaissance.

If a customer decides to use their on-premises DNS server for third-level domains, even when distributed across different public resolvers, these resolvers will still initiate recursive queries to the customer's on-premises server. This situation could lead to a decrease in service quality or even denial-of-service due to the bottleneck effect. However, it is impossible to completely prevent DNS communication with the target server when attempting to validate the results of passive subdomain gathering. The best course of action is therefore to implement rate limiting and use public resolvers to mitigate the impact.

For all the above reasons, the activity cannot be considered entirely passive. In terms of DNS reconnaissance, passive reconnaissance involves gathering information from open sources while resolving/subdomain brute-forcing is categorized as active reconnaissance. We are making an exception in this case, however, as we are required to define a scope and provide the customer with a Letter of Authorization outlining the future scope of the work before starting the project.

We therefore consider this type of reconnaissance to be semi-passive and acceptable under certain circumstances. Of course, it is crucial to always coordinate such activities with the client.

DNS AXFR query versus DNSSEC status tests

A misconfigured DNS server vulnerable to zone transfer and an NSEC-signed zone (instead of an unsigned or NSEC3-signed zone) both can allow attackers to dump all subdomains in a DNS zone. Probing AXFR is considered active while probing the DNSSEC status is not. DNS zone walking is still considered active.

This is because querying the DNSSEC status is part of normal internet traffic and the fact that, after the query (probing), knowing whether the zone could be dumped or not is only a natural extension of the result. Meanwhile, a DNS AXFR request is highly abnormal for unauthorized users.

Versioning

In general, it is the action itself — and not what information can be gathered from the action — that matters in passive versus active reconnaissance. Attempting to run an exploit to see whether a web app is vulnerable is considered active reconnaissance, but often checking the version number at the footer of the website will tell you whether it is vulnerable or not. The “check” function in many Metasploit exploits only checks the version number of the target. Reading the version number displayed on a web app cannot be considered active reconnaissance, but many web apps try to conceal their version, so more advanced versioning methods are required.

Versioning could get noisy as you crawl pages or probe API endpoints to try to spot the difference in response and narrow down a version range. This is probably already within the realm of active reconnaissance. As a side note, automated tools with passive versus active switches should used on the side of caution when it comes to versioning. For example, by default (in passive mode) WPScan will not even try to access the README file for versioning.

5.

RECONNAISSANCE REQUIREMENTS FOR CYBERSECURITY ENGAGEMENTS

Reconnaissance is a process involving recursive discovery, that is using one piece of information to find other pieces of information. When we obtain new information, the process starts again and repeats until no new information emerges. Depending on the type and stage of engagement, the information of interest may vary, but the general thought process remains the same.

For nearly all types of engagements, the focus is on publicly accessible assets owned or operated by the target. This encompasses domains, subdomains, owned IP ranges, ASNs, VPS hosts (IP addresses), cloud storage buckets, and cloud tenant subdomains, among others. Before an engagement begins, it is customary to draft a Letter of Authorization (LOA) for the potential customer. At this stage, only passive reconnaissance methods can be used as we seek to identify assets owned by the customer on the Internet. Ownership details can be confirmed through registration information such as whois records, third-party sources like Google, or associations like links, shared hosting IP, and shared TLS certificates.

When the LoA is signed, permission is granted to conduct active reconnaissance. This includes activities such as subdomain brute-forcing, virtual host enumeration, directory brute-forcing, and port scanning, and all these steps are taken against assets as authorized by the LoA.

Penetration Testing:
Revealing vulnerabilities with precision

Penetration tests require a meticulous approach to reconnaissance. In this corporate ballet, efficiency is key. Threat actors opt for simplicity, seeking vulnerabilities that provide the quickest access. Techniques like subdomain brute-forcing, directory brute-forcing, versioning for application vulnerabilities, and login brute-forcing are precise steps to uncover weaknesses. Apart from these, leaks of source code, secrets, credentials, and internal documentation add depth to the investigation. While social engineering might be off-limits, controlled reconnaissance among staff can still yield valuable insights.

Web Application Vulnerability Assessments: A methodical approach

When it comes to web application vulnerability assessments, the reconnaissance approach is measured and clear. Targets are well-defined, usually in the form of subdomain names. The objective is to identify vulnerabilities within the application using a methodical approach. Possible steps include directory enumeration, scrutiny of public source code, and gathering insights into API endpoints. Unlike the intense movements involved in penetration tests, subdomain brute-forcing involves taking a backseat and the focus narrows down to the application's specific security features.

Red Teaming: The comprehensive strategy

Building on the foundation of penetration testing, red teaming widens the scope of a reconnaissance strategy. It includes third-party SaaS platforms like Jira, Confluence, and Salesforce — integral elements in the corporate playbook.

Personal reconnaissance turns to understand employees better, making sure credentials are safe, and guarding against social engineering tricks. With active reconnaissance, however, you must tread lightly, maintaining a noise level no higher than the usual background of threat actors on the internet. Strict rate limits and geographically suitable IP addresses act as silent partners. In certain scenarios, an in-depth reconnaissance unfolds and exposes vulnerabilities in specific subjects — through a targeted spear phishing attack or a dependency confusion attack.

6. PERFORMING RECONNAISSANCE: WHY SHOULD YOU FOLLOW A STANDARD OUTPUT FORMAT?

The results of reconnaissance can be presented in various formats, and the format of output data will generally depend on the task. Below is one of the possible structures of output data used during the passive reconnaissance stage. It is later included in the Letter of Authorization (LoA) and supplemented further during the active reconnaissance process. The format of output data does not necessarily have to follow the mentioned format, but sticking to a specific structure is useful when working in a team because it makes for more efficient collaboration and helps to analyze output data more effectively.

domains.txt	• Domains owned or operated by the organization (information determined with a high level of confidence) • One domain per line • Machine-readable plain text file • Optionally grouped by function and sorted roughly by priority
subdomains.txt	• Subdomains of the domains listed in domains.txt • One subdomain per line • Machine-readable plain text file • Subdomains operated by the organization but not under domains owned by the organization (e.g., ORG.s3.amazonaws.com) should NOT be included • Only subdomains that have an IPv4 or IPv6 address should be included; subdomains that only have a TXT record (e.g., _dmrac.) should NOT be included • If a domain zone is a wildcard, include *.wildzone.example.com in the file; optionally remove all discovered subdomains in this zone if they are unlikely to be real (for example, passive reconnaissance tools will often report nonexistent subdomains like 3.11.34.56.wildzone.example.com for wildcard zones) • Sorted by the order of domains as they appeared in domains.txt, then for each domain, by order of the string comparison after reversing the field separated by dots
ips.txt	• XaaS assets used by the organization, but not directly owned or operated (self-hosted), for example, S3 bucket, Azure AD tenant, GitHub account, Jira, Confluence, Salesforce • One URL per line and loose format
result.txt	• Copy-paste all results from domains.txt, subdomains.txt, ips.txt, and xaas.txt into sections • Include IP ranges operated by the organization (if determined with a high level of confidence) • Optionally include additional comments that are not included in the machine-readable files • Optionally include additional information such as leaks and email patterns

subdomains-ip.txt	• Machine-readable and tab-separated CSV plain text file: the first column is the subdomain, the second column is the IP address that the subdomain resolves to, and the third column is the ASN information relating to the IP address • Sorted by the order of subdomains as they appear in subdomains.txt, then for each subdomain, by the order of the IP addresses, with IPv4 first
ips-subdomain.txt	• Machine-readable and tab-separated CSV plain text file: the first column is the IP address, the second column is the subdomain that the IP address resolved from, and the third column is the ASN information relating to the IP address • Only included IP addresses operated by the organization (as determined with a high level of confidence and listed in ips.txt); generated by swapping the subdomain and IP column of *subdomains-ip.txt*, then stable sort (preserve the subdomain column order) by the IP column, delete IP addresses that are not operated by the organization (e.g., public hosting, CDN, proxy) • Optionally include other IP addresses that are owned or operated by the organization but do not have an associated subdomain (if attributed with a high level of confidence)

Others

For pentesting/red teaming purposes, linking employees' corporate accounts to their personal accounts exposes them to potential vulnerabilities.

One such conduit is password reuse providing a gateway for initial access to the target network. Additionally, it becomes crucial to examine Software as a Service (SaaS) solutions and code repositories like GitHub to identify potential leaks that could expose sensitive information.

- email.txt or username.txt: Email addresses and usernames at the organization could follow a pattern (a verification column should be included)
- personnel.txt: Personnel working at the organization with name, position, work email address, personal email address, LinkedIn profile, etc.

7. STEPS FOR PASSIVE AND ACTIVE RECONNAISSANCE

After discussing the components of reconnaissance, its applicability, and the desired outcomes, it's time to describe the steps required to achieve those results. As mentioned, the reconnaissance process is recursive, so it might make sense to return to passive information gathering after performing active reconnaissance until the desired outcome is satisfactory.

Passive reconnaissance domains.txt Manual browsing

1. **Gain a good understanding** of the organization's structure from Wikipedia, the org chart on the organization's official website, news items, financial disclosure documents, and other OSINT sources. Determine horizontal (e.g., departments) and vertical (e.g., regional offices) organizational structures. All horizontal targets will usually be included, but the extent to which vertical targets are included should be discussed. Aspects to consider in this respect include the number of vertical targets (perhaps there are too many) and how "close" they are to the parent organization from an operational point of view (e.g., being a majority shareholder does not necessarily imply being involved in day-to-day operations).
2. **Browse social media.** Look for links, QR codes, projector screens, computer screens, sticky notes, etc. Any information that might be helpful to know in an attack (e.g., what software is running on employee computers) is valuable.
3. **Navigate the organization's website manually.** Focus on the header, footer, sidebars, sitemap, and employee usernames and emails.

Index browsing

1. **Google & Bing search.** Use advanced search operators to subtract already discovered domains. Bing sometimes works better with the "site:" operator. Some useful queries:
    - <ORG-NAME>
    - <ORG-NAME-NATIVE-LANG>
    - <ORG-NAME> -site:<ORG-MAIN-DOMAIN>
    - <ORG-NAME-NATIVE-LANG> -site:<ORG-MAIN-DOMAIN>
    - inurl:<ORG-NAME> -site:<ORG-MAIN-DOMAIN>
    - inurl:<ORG-NAME-NATIVE-LANG> -site:<ORG-MAIN-DOMAIN>

Repeat the above several times but replace the organization name with the department name, regional office name, subsidiary, etc. If the sub-unit's name is not unique, add the organization name back in the query.

Using many advanced search operators is sometimes unreliable. For example, searching for site:group-ib.com -site:www.group-ib.com might exclude more subdomains than you asked for (other than www.)

2. **host.io for finding co-host, link to, and link back.**
    - Append all domain names (one on each line) with “https://host.io/”. Copy this URL list and open all using the Bulk URL Opener extension.
    - Manually browse any promising domains in the result pages to determine ownership.
    - Keep the following in mind:
        - Hosting on an IP address within a non-public ASN (e.g., not ISP, AWS) is a strong indicator of ownership.
        - Co-hosting on an IP address that is not public hosting is a strong indicator of ownership.
        - The same authoritative DNS server on a non-public (e.g., not ISP, Cloudflare) authoritative DNS server is a strong indicator of ownership.

3. **viewdns.info for finding whois and DNS-related domains.**
    - Append all domain names (one on each line) with “https://viewdns.info/reversewhois/?q=”. Copy this URL list and open all using the Bulk URL Opener extension.
    - Append all non-public authoritative DNS servers with “https://viewdns.info/reversens/?ns=”. Copy this URL list and open all using the Bulk URL Opener extension.
    - Results are often outdated and incomplete, but doing so is still worth a try.

4. **Different TLD same name**
    - If the results so far include many cases of the same name in different TLDs (e.g., group-ib.com, group-ib.ru, and group-ib.us are all registered), it might be worth doing a TLD search. Command:

```
dnsrecon -t tld -d group-ib.  (dnsrecon - https://github.
com/darkoperator/dnsrecon)
```

5. **Different TLD same name**
    - Looks like: (^|.)group-ib\.com$
    - Update from domains.txt as more domains are found. Make a final version before moving into subdomain discovery.
    - Use text editor replace function, or:

```
sed -e 's/^/(^|.)/' -e 's/$/$/' -e 's/\./\\./g' domains.txt
| tee domains-regex.txt
```

subdomains.txt

For passive subdomain discovery, we mainly rely on automated aggregators.

1. **Amass enum** (amass - https://github.com/owasp-amass/amass)

```
amass enum -passive -d $(paste -s -d , domains.txt) | tee
amass.txt
```

2. **Subfinder** (subfinder - https://github.com/projectdiscovery/subfinder)

```
subfinder -all -dL domains.txt | tee subfinder.txt
```

3. **DNS PTR record**

```
dnsrecon -r <IP-RANGE>
```

- From experience, this method can often be outdated and unfruitful, but it is still worth checking whether there is a range of IPs, especially owned ASNs.

4. **DNSSEC NSEC check**

   https://dnssec-debugger.verisignlabs.com/

- If the target zone does not have DNSSEC, it is not vulnerable to zone walk. If the target zone is NSEC-signed, zone walk can directly dump all subdomains. If the target is NSEC3-signed, zone walk can dump the salted hash of all subdomains, which allows for offline subdomain brute-forcing.
- Do NOT zone walk without permission. Zone walking probably counts as active reconnaissance.

5. **(Optional) Google & Bing search**

- See above for query techniques
- Usually not necessary as the results are likely to be covered by other tools already.

6. **Combine and filter results**

```
cat darkmatter.txt subfinder.txt amass.txt > tmp.txt
while read -r domain; do rg $domain tmp.
txt | awk -F '.' '{for (i=NF;i>=1;i--) printf
"%s%s", $i, (i==1 ? "\n" : FS)}' | sort -t .
-k1,1 -k2,2 -k3,3 -k4,4 -k5,5 -k6,6 -k7,7 -u | awk -F '.'
'{for (i=NF;i>=1;i--) printf "%s%s", $i, (i==1 ? "\n" :
FS)}'; done <domains-regex.txt | tee subdomains.txt
dnsx -r resolvers-trusted.txt -l subdomains.txt | tee tmp.
txt
```

```
while read -r domain; do rg $domain tmp.
txt | awk -F '.' '{for (i=NF;i>=1;i--) printf
"%s%s", $i, (i==1 ? "\n" : FS)}' | sort -t .
-k1,1 -k2,2 -k3,3 -k4,4 -k5,5 -k6,6 -k7,7 -u | awk -F '.'
'{for (i=NF;i>=1;i--) printf "%s%s", $i, (i==1 ? "\n" :
FS)}'; done <domains-regex.txt | tee subdomains.txt
```

(dnsx - https://github.com/projectdiscovery/dnsx)

- Use a list of trusted public resolvers. We recommend: https://github.com/trickest/resolvers/blob/main/resolvers-trusted.txt

7. **(Optional) Wildcard subdomain detection**

```
sed -r 's/^[^\.]+\.//' subdomains.txt | sort -u | sed 's/^/
x348tfghasdikuhqaf./' | awk -F'.' 'NF!=2' | tee tmp.txt
dnsx -r resolvers-trusted.txt -l tmp.txt | tee wild.txt
```

8. **Make a temporary subdomains-regex.txt that contains the regex version of the subdomains for later use in matching subdomains-ip.txt:**

```
sed -e 's/^/^/' -e 's/$/\\t/' -e 's/\./\\./g' subdomains.
txt | tee subdomains-regex.txt
```

subdomains-ip.txt

Simply query IP and ASN information and then reformat it.

1. **dnsx -l subdomains.txt -resp -a -aaaa -asn | tee tmp.txt**

```
sed -i -e 's/  / /' -e 's/ \[/\t/g' -e 's/]//g' tmp.txt
while read -r subdomain; do rg $subdomain tmp.txt | rg ':'
-v | sort -t \t -k2 -s -V | tee -a subdomains-ip.txt;
rg $subdomain tmp.txt | rg ':' | sort -t \t -k2 -s | tee
-a subdomains-ip.txt; done <subdomains-regex.txt
```

2. **ips-subdomain.txt and ips.txt**

Start with IPs resolved from subdomains and filter out IPs not owned by the organization.

```
rg ':' -v subdomains-ip.txt | awk -F'\t'
'{OFS="\t"; temp=$1; $1=$2; $2=temp; print}' | sort
-t $'\t' -k1,1 -V -s | tee -a ips-subdomain.txt; rg ':'
subdomains-ip.txt | awk -F'\t' '{OFS="\t"; temp=$1;
$1=$2; $2=temp; print}' | sort -t $'\t' -k1,1 -s | tee
-a ips-subdomain.txt
```

Identify domains resolving to CDN IPs. These should not be included in ips-subdomain.txt or ips.txt, but we can include buckets in xaas.txt.

3. **dnsx -l subdomains.txt -resp -a -aaaa -cdn | tr -d '\n' | tee tmp.txt**

```
sed -i -e 's/  / /' -e 's/ \[/\t/g' -e 's/]//g' -e  tmp.txt
sed '/ $/d' tmp.txt | cut -d $'\t' -f 2 | sort -u | sed
-z '$ s/\n$//' | tee cdn-ips.txt
rg -F -f cdn-ips.txt -v ips-subdomain.txt | sponge
ips-subdomain.txt
```

After filtering out the usual suspects like Cloudflare and AS0 local addresses, you must manually verify the ownership of the remaining IP addresses. First, look for semi-continuous blocks of IP addresses. These might be IP ranges or even Autonomous System Numbers (ASNs) owned by the organization. Usually, large IP ranges, and especially ASNs, give clear ownership information in whois records. You can also look for the organization's name plus "ASN" or "whois" in Google search. Add any such IP ranges into result.txt. For the remaining IP addresses, the most definitive answer would be a whois record showing the organization as the owner or providing an admin contact email, but these records are often redacted. In such cases, you can check the IP address on https://host.io/ip/. If there are a lot of irrelevant co-hosting domains, the IP address should be filtered out.

xaas.txt

1. **Buckets**
   - gobuster s3 (gobuster - https://github.com/OJ/gobuster)
   - Group-IB cloud-sherlock (https://github.com/Group-IB/cloud_sherlock)
2. **Azure tenant**
   - https://gettenantpartitionweb.azurewebsites.net/
3. **BaaS**
   - Subdomain CNAME or redirect
   - Network traffic when browsing webapp
   - Scan APK with APKLeaks
4. **SaaS**
   - Subdomain CNAME or redirect
   - Manually browsing the official website
   - Google search

Others

1. **Leaks**
   - Open source repos: secrets, vulns, internal resources
   - Google Dorking
   - Leakix (https://leakix.net/)

2. **Personnel**
   - LinkedIn
   - Official websites
   - Email patterns

Using solutions like Shodan and Censys can be beneficial at any stage of passive reconnaissance, so we recommend not overlooking such tools.

3. **Active reconnaissance**

After receiving authorization, you can start active reconnaissance. As mentioned in the beginning, different engagements require different reconnaissance, but some requirements are universal.

4. **Subdomain brute-forcing**
   - gobuster dns
   - zdns (zdns - https://github.com/zmap/zdns)
   - Wordlist: one of https://github.com/danielmiessler/SecLists/tree/master/Discovery/DNS
   - httpx can be used to filter out interesting subdomains in a wildcard zone (httpx - https://github.com/projectdiscovery/httpx)

5. **DNS zone transfer**
   - dig AXFR

6. **DNSSEC NSEC zone walk**
   - dnsrecon

7. **Virtual host enumeration**
   - gobuster vhost
   - Wordlist: one of https://github.com/danielmiessler/SecLists/tree/master/Discovery/DNS

8. **Directory brute-forcing**

   - dirsearch or gobuster
     (dirsearch - https://github.com/maurosoria/dirsearch)
   - Quick wordlist:
     https://github.com/Bo0oM/fuzz.txt/blob/master/fuzz.txt
   - Wordlist: -
     https://github.com/danielmiessler/SecLists/tree/master/Discovery/Web-Content

9. **Port scanning**

   - Quick scan for open ports: nmap, rustscan
     (nmap - https://nmap.org/;
     rustscan - https://github.com/RustScan/RustScan)
   - Detailed scan with scripts: nmap

10. **Vulnerability scanning**

    - Nuclei (nuclei - https://github.com/projectdiscovery/nuclei)

11. **Technology identification and versioning**

    - httpx
    - gowitness or aquatone takes screenshots
      (gowitness - https://github.com/sensepost/gowitness ;
      aquatone - https://github.com/michenriksen/aquatone)
    - Wappalyzer browser extension
      (wappalyzer - https://www.wappalyzer.com/)
    - Manual: This might be as simple as checking the footer
      or as involved as downloading 10 versions locally and comparing them with the target.

8. CHECKLIST AND PRIORITIES

Given the extensive scope of the process, it is best to track progress in tables and prioritize targets that will most likely yield results.

A convenient aspect of CSV is that it is simple enough to be easily manipulated using CLI tools, but it also works well with spreadsheet software. You can enhance subdomains-ip.txt with additional columns from active reconnaissance (mainly technology identification), imported to spreadsheet software, and work as a to-do list of web applications to be examined. You can enhance ip-subdomains.txt with port scanning results, imported to spreadsheet software, and work as a to-do list for host scanning.

Specialists are advised to test the following types of targets first: targets that are more likely to be vulnerable, targets that require less effort to test, targets that are easier to exploit, and targets with higher impact. These properties cannot be known for sure without first testing the target, but some heuristics are good approximations and help prioritize targets at a glance:

- Complex is more vulnerable than simple
- Older software is more vulnerable than newer software
- Low-traffic (e.g., employee portal) sites are more vulnerable than high-traffic sites
- Non-privileged sites are easier to test than privileged sites
- Privileged sites have a higher impact than non-privileged sites
- Widely used software is easier to test and exploit than bespoke software
- Tailored software is more vulnerable than widely used software
- Infrastructure has a higher impact than individual business logic

As some of these priorities are conflicting, prioritization does not follow a specific protocol. The goal is to avoid challenging situations like trying to find a zero-day vulnerability in GitLab before first attempting login brute-forcing. There might also be other priorities to consider. For example, stealth is important in a red team engagement, so we should first test things that are less likely to be noticed (e.g., slow credential stuffing).

9. THE AFTERMATH OF RECONNAISSANCE: SHIELD AGAINST INTRUSIONS WITH GROUP-IB

Following reconnaissance and gaining insights into potential entry points and system vulnerabilities, organizations can implement substantial measures to prevent and protect against unauthorized intrusions into their networks. Group-IB recommends a combination of proactive and reactive approaches to address both active and passive reconnaissance, handled either internally or with the support of experienced third-party service providers.

For passive reconnaissance, maintaining awareness is key. Companies can conduct self-reconnaissance to have an accurate understanding of their attack surfaces and ensure that their employees have the skills to defend the company effectively. Group-IB's Attack Surface Management (ASM) helps to proactively manage external attack surfaces and swiftly identify vulnerabilities, thereby reducing the risk of unauthorized entry points that compromise system integrity.

Automated analysis is crucial but should complement manual analysis for a more in-depth understanding. Group-IB's experts conduct Red Teaming exercises that involve realistic attack simulations — the goal is to emulate the tactics, techniques, and procedures used by potential adversaries and assess a system's or an organization's resilience. As part of Group-IB's comprehensive offensive security services, reconnaissance ("recon") plays an integral role in helping to sort defenses against real-world threats.

In the case of active reconnaissance, early indications can be spotted through log reviews and the use of intrusion detection systems (IDSs) to detect anomalies in network or traffic activity. The Network Detection and Response System (NDR) module under Group-IB's Managed Extended Detection and Response (MXDR) continuously monitors networks, actively detects intrusions, and alerts teams so that they can respond swiftly.

For security incidents or drill sessions, a well-thought-out incident response plan is equally essential. This plan guides organizations in recognizing, responding to, and mitigating incidents so that normal activities can resume swiftly. All employees should be trained in and have access to the procedure to ensure a coordinated response.

Group-IB's Digital Forensics and Incident Response (DFIR) experts are available to assist in devising the right incident response plan tailored to your organization's needs.

10. CONCLUSION

In the constantly evolving space of cybersecurity — and reconnaissance specifically — techniques are in a state of constant flux. Intrusion methods are depreciating fast while new ones rise to prominence... even as this information piece is being written.

Changes to the digital space and the increasing number of communication channels keep equipping attackers with new and powerful tools. This whitepaper serves as a base-level guide on active and passive reconnaissance, and it may not cover all the emerging methods of performing reconnaissance in the current threat landscape. The information contained herein can be leveraged by security teams for understanding and performing automated and manual security analyses through reconnaissance. This article mainly discusses manual reconnaissance for uncovering an organization's or company's assets, with a limited focus on gathering information about individuals.

While not claiming to be exhaustive, this guide provides basic information about various types of reconnaissance, their objectives, how to execute them, and crucial considerations to bear in mind.

Do you need help with mastering reconnaissance to stop intrusions and safeguard your network?

Get in touch with Group-IB experts ↗

Who we are: overview

Group-IB's audit and consulting technology and services have helped companies in various industries to identify critical vulnerabilities in their external and internal network perimeters.

Our range of cutting-edge services include:

Penetration Testing

Red Teaming

Compliance Audit and Consulting

Security Assessments

Audit process by Group-IB

Our success lies in our established three-step process.

1. Preparation

- Planning work
- Collecting initial data
- Analyzing internal documentation

2. Examination and analysis

- Conducting interviews
- Collecting audit evidence
- Analyzing the data collected

3. Report Generation

- Preparing a guided report
- Drafting recomendations on how to eliminate incosistencies

Established Industry Expertise

Group-IB has been a trusted partner for businesses worldwide as regards security compliance, with over 50 critical compliance checks performed every year.

Every member of our consulting team is a certified auditor with extensive experience in compliance auditing for corporations in fields such as healthcare, financial services, critical services, and manufacturing.

About Group-IB

Group-IB is a leading creator of technologies designed to investigate, prevent, and fight cybercrime.

1,400+
Successful investigations of high-tech cybercrime cases

250+
employees

650+
enterprise customers

60
countries

$1 bln
saved by our client companies through our technologies

#1*
Incident Response Retainer vendor

120+
patents and applications

17
inventors in our team

4
Digital Crime Resistance Centers (Singapore, Dubai, Amsterdam, Phuket)

* According to Cybersecurity Excellence Awards

Global partnerships

INTERPOL

Europol

Recognized by top industry experts

FORRESTER®

kuppingercole ANALYSTS

Gartner

IDC

FROST & SULLIVAN

Preventing and investigating cybercrime since 2003

FIGHT AGAINST CYBERCRIME

GROUP-IB.COM
INFO@GROUP-IB.COM

APAC
+65 3159 4398

EU & NA
+31 20 226 90 90

MEA
+971 4 568 1785