Chaining in action: techniques, terminology, and real-world impact on business

BY ELEANOR BARLOW · FEBRUARY 16, 2026

What you will learn in this blog

- **What chaining is** and how combining lower-severity issues can create a high-impact security risk.
- **Key chaining techniques and terminology**, such as pivoting, lateral movement, and privilege escalation.
- **How chaining is identified and prioritized in practice**, including the role of PTaaS and how researchers can use chaining to uncover critical attack paths and guide next steps for business prioritization.

What is chaining?

The term chaining, which is also used interchangeably with the term exploit chaining, expresses the process of identifying and combining two or more lower-impact vulnerabilities to form an attack with high impact, such as those leading to direct access to a server.

When alone, a single minor vulnerability does not pose an instant threat, but when chained together with other vulnerabilities, it can create a critical impact that can have devastating impacts, for example, in the form of data exfiltration and even full account compromise.

“Companies can often overlook the impact of exploit chaining, especially if they only factor in one finding at a time”- **Takashi Doyama, Penetration Testing Delivery Manager, Intigriti**

And chaining can often lie undetected, as a single security tool may not recognize or be able to link together the full threat at play.

“Scanners tend to miss chained issues because the final step is often hidden by some prerequisite path or delivery mechanism.” – **Alex Olsen, Hacker Community Lead, Intigriti**

What is an example of chaining that could be found in your environment?

Say there are two separate vulnerabilities that are both different in nature but are on the same server. The first vulnerability could be something like an arbitrary file read, which is when a user is able to read files on a server’s file system that they should not be able to access or view. If a vulnerable web application could be manipulated in a certain way, it could be instructed to read text files within a system that was not intended to be accessed. For instance, a threat actor, using arbitrary file read, could read a list of usernames within a Linux server, contained within the /etc/passwd file.

“Combining this with the second vulnerability, such as a weak password policy, this threat actor could conduct a brute-force attack against the server itself, successfully obtaining the password and gaining direct command-line access to the server. And although these two types of vulnerabilities are separate in nature, when combined in the form of an exploit chain, they could allow a threat actor with far more access compared to if these vulnerabilities were exploited individually.” – **Takashi Doyama**

One could argue that the usefulness of arbitrary file read is limited if there is no data present that is attractive to a threat actor. And that weak password policies would be difficult to exploit via brute-force attacks if valid usernames were not known.

But as Takashi explains, “Like pieces in the puzzle, if these vulnerabilities were combined, it would allow the threat actor to gain direct access to the server itself, which makes both these findings incredibly serious.”

Important chaining-related terminology

What is a privilege escalation?

Privilege Escalation is the term used when a threat actor moves up the levels of permissions. For instance, a vertical escalation can be where someone with access to user rights gains access to admin rights. A horizontal escalation is where someone with user access gains access to another user’s access. The threat actor can even move from guest to user, all the way to admin, to full system access, in an exploit chain, gaining access to each account in succession.

What is lateral movement?

Lateral movement is where the user moves from one compromised system to another within the network. Once one machine is compromised, others can be infected. For instance, access to one workstation can lead to access to the file server, which can lead to access to the domain controller.

What is pivoting?

Pivoting is when a compromised system is used as a stepping stone to reach other compromised systems on different parts of an internal network. This can also be referred to as ‘daisy chaining’ or ‘pivot attacks’.

‘In a network hack, which is a hack against internal networks, it's all about pivoting, which is hacking from one server to another to get to important servers, which is similar to exploit chaining, but here the threat actor covers the whole network. So, they could, theoretically, find out one password and spray to other servers, or try and attack harder-to-access servers by channeling network traffic via another internal server that they have compromised.” - **Takashi Doyama**

The role of PTaaS in identifying chaining

The example provided above, regarding weak password policies, is almost always down to misconfigurations. Exploit chaining and its dangers are a good reason why PTaaS is important in this instance, because in a single report, you can get an overall context of how each vulnerability fits with the others.

“When a threat actor gains direct command line access to a server, they have access to the server in the same way that a user has access to their laptops. Combined, this has a critical impact because an attacker could then go and directly manipulate the server itself and have greater control, so they could read any files, write malware onto the system, and even figure out if they can get full root access to the server, or conduct attacks from the server they've just hacked. PTaaS would need to be used to identify these misconfigurations.”- **Takashi Doyama**

Be it critical, high, medium, low, or informational, there are two methods to judge the severity of a vulnerability.

- The first is via the CVSS3.0 / CVSS4.0 string, which gives an overall synopsis of the vulnerability.
- The second is a subjective decision from the author of the report. Skilled penetration testers are highly valuable as they factor in context in which the vulnerability was identified.

“Subjectiveness is very important here, because although these two vulnerabilities could be classed as high-risk vulnerabilities on CVSS 3.0 alone, given that you could chain arbitrary file read with a weak password policy to get direct access, a pentester could class both vulnerabilities as critical, as they were both used to hack into the server.” - **Takashi Doyama**

How do researchers use chaining to identify high-severity impacts?

From a Bug Bounty Program perspective, there is a shift in mindset from looking at bugs as solely singular issues, as many traditional forms of security testing do, to looking at them as chains of bugs and analyzing what the overall impact would be if they were connected.

“Every bug, or quirk, or vulnerability could be a stepping stone, and the real impact could be a little bit further ahead. We, the researchers, should also think about looking sideways or horizontally. So, a low or informational finding in service ‘A’ could unlock something critical in service ‘B’, for example. And I think this is more common when we're dealing with things like microservices or an application that calls out to many different places.” – **Alex Olsen**

Anything that the target implicitly trusts should also be tracked. Unlike automated scanners, researchers can do this and keep a lookout for elements that question trust boundaries, such as IPs on the same range, subdomains, or S3 buckets. It's also important for researchers to analyze what can expand the attack surface.

“For instance, a server-side request forgery, open redirects, cross-site scripting, calls with wildcards, hard-coded tokens and secrets, all sorts of things can be used to expand the attack surface. And when we're testing, we should be building a mental library of these issues so that we can mix and match until something juicy clicks into place.”- **Alex Olsen**

At Intigriti, we host monthly web-based ‘Capture The Flag’ (CTF) challenges to engage the security researcher community. A recent challenge drew inspiration from the Marvel Cinematic Universe, specifically Thanos' quest to collect all six Infinity Stones. The challenge required researchers to chain multiple client-side vulnerabilities across different subdomains to ultimately achieve XSS on the main challenge page. This blog demonstrates advanced techniques for exploiting XS-Leak vulnerabilities, post-message handlers, and various browser APIs.

Intigriti's triage standards

When vulnerabilities are part of a chain, meaning one vulnerability enables or escalates another, the combined impact will be considered if the vulnerabilities are considered unique and not previously reported. Each vulnerability within the chain that is independently fixable may be tracked as a different issue and subsequent findings referencing the same behavior may partially be considered a duplicate if the newer finding does not show additional impact. If the new finding does show additional impact, the difference between the impacts of the two submissions should be considered for the new finding.

For more information, read Intigriti's triage standards.

Next steps

Identifying potential vulnerability chains is critical for businesses wanting to avoid serious security escalation later down the line. By using the power of the crowd, Intigriti is best placed to provide insights into how attackers think and operate, and to identify potential chaining paths in your environment early on, long before exploitation.

By adopting a layered approach to your proactive security strategy, Bug Bounty, combined with PTaaS, can be used to systematically uncover technical weaknesses that could form a vulnerability chain across applications, so that threats are not looked at in isolation.

If you found this blog insightful, have questions, or would like to speak to one of our experts, contact the team today.

AUTHOR

Eleanor Barlow

Eleanor Barlow is a London-based Senior Cyber Security Technical Writer at Intigriti, with 9+ years' experience reporting on and writing for the cyber and tech sector. She specializes in data-driven content on cybersecurity and bug bounty intelligence, helping organizations benefit from the latest trends and insights.