Understanding the Status and Strategies of the Code Signing Abuse Ecosystem

Hanqing Zhao, **Yiming Zhang**, Lingyun Ying, Mingming Zhang,

Baojun Liu, Haixin Duan, Zi-Quan You, Shuhao Zhang

清華大學 Tsinghua University

奇安信 QI-ANXIN

ZGC LAB

NDSS SYMPOSIUM

Surge in Software Supply-Chain Attacks

- Supply-chain attacks have become frequent, threatening software security.

1

Kaseya attack (2021):

- Attackers **impersonated a legitimate vendor** and delivered **REvil** ransomware via an update
- ~1,500 organizations were affected, mistakenly installing/executing malicious code

2

NotPetya attack (2017):

- NotPetya was distributed through a compromised **M.E.Doc** update mechanism
- Victims’ machines **executed a tampered update**, causing >$10B in economic losses

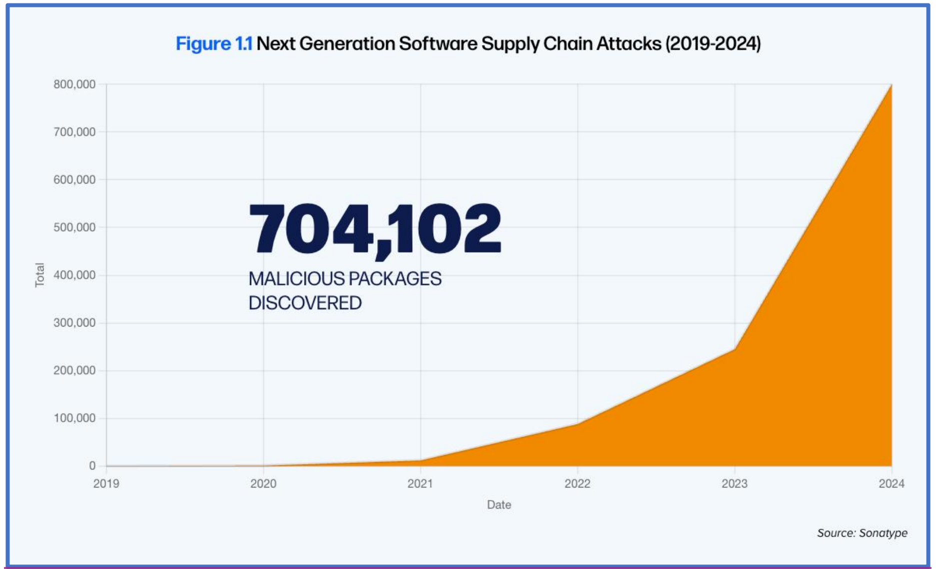


Figure 1.1 Next Generation Software Supply Chain Attacks (2019-2024)

Source: Sonatype

Global Software Supply-Chain Attack Statistics (2019–2024)*

* *https://www.sonatype.com/hubfs/SSCR-2024/SSCR_2024-FINAL-10-10-24.pdf*

Surge in Software Supply-Chain Attacks

- Supply-chain attacks have become frequent, threatening software security.

1

Authenticity: Software provenance is hard to verify

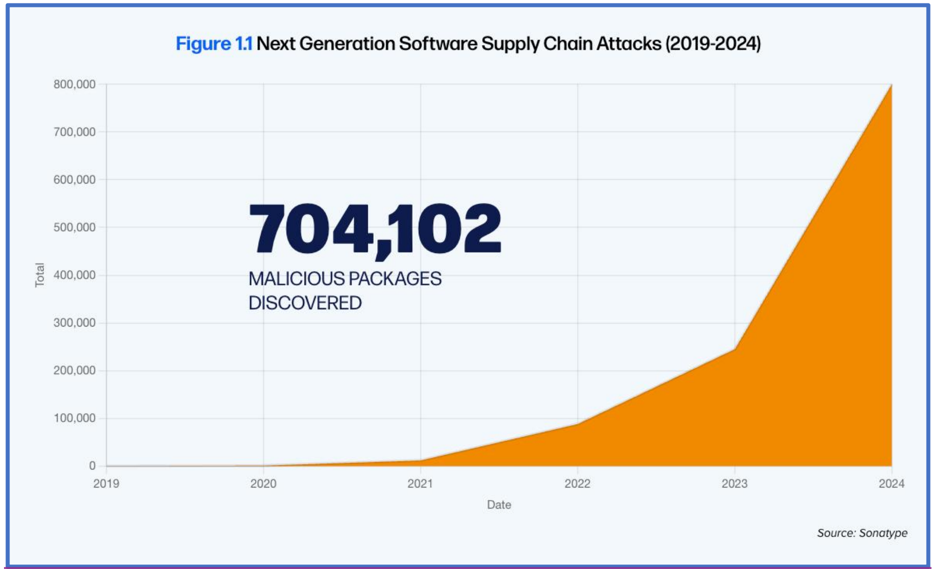
Kaseya attack (2021):

- Attackers **impersonated a legitimate vendor** … te.
- … ed, mistakenly installing/executing malicious code.

2

NotPetya attack (2017):

- NotPetya was distributed through a compromised **M.E.Doc** update mechanism
- Victims’ machines **executed a tampered update**, causing >$10B in economic losses



Global Software Supply-Chain Attack Statistics (2019–2024)*

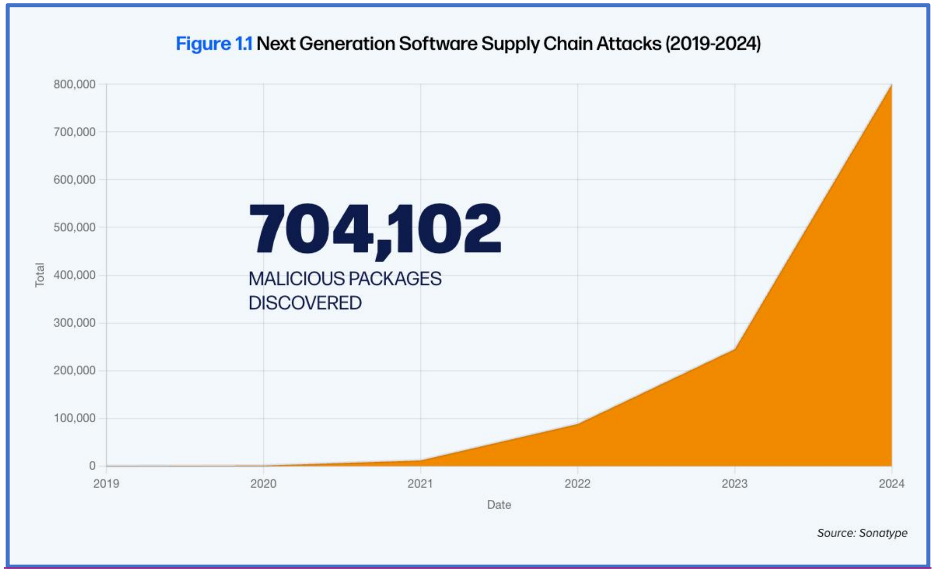
* *https://www.sonatype.com/hubfs/SSCR-2024/SSCR_2024-FINAL-10-10-24.pdf*

Surge in Software Supply-Chain Attacks

- Supply-chain attacks have become frequent, threatening software security.

1 **Authenticity: Software provenance is hard to verify**

2 **Integrity: Software can be tampered with**

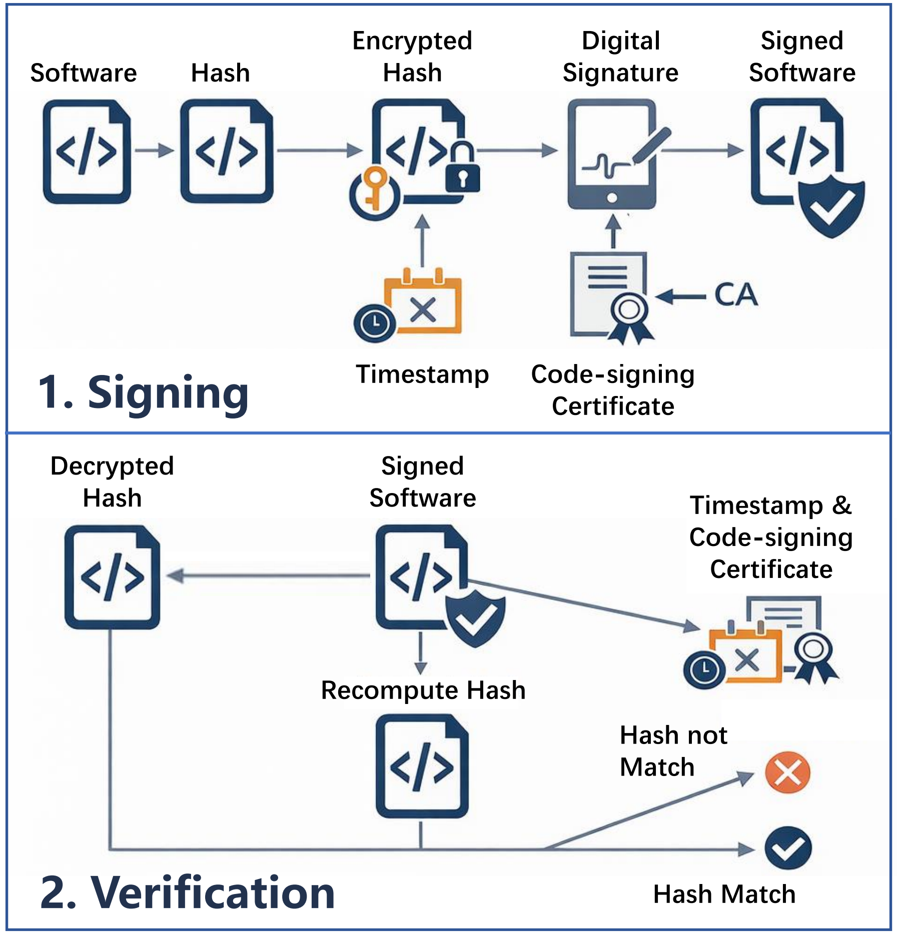


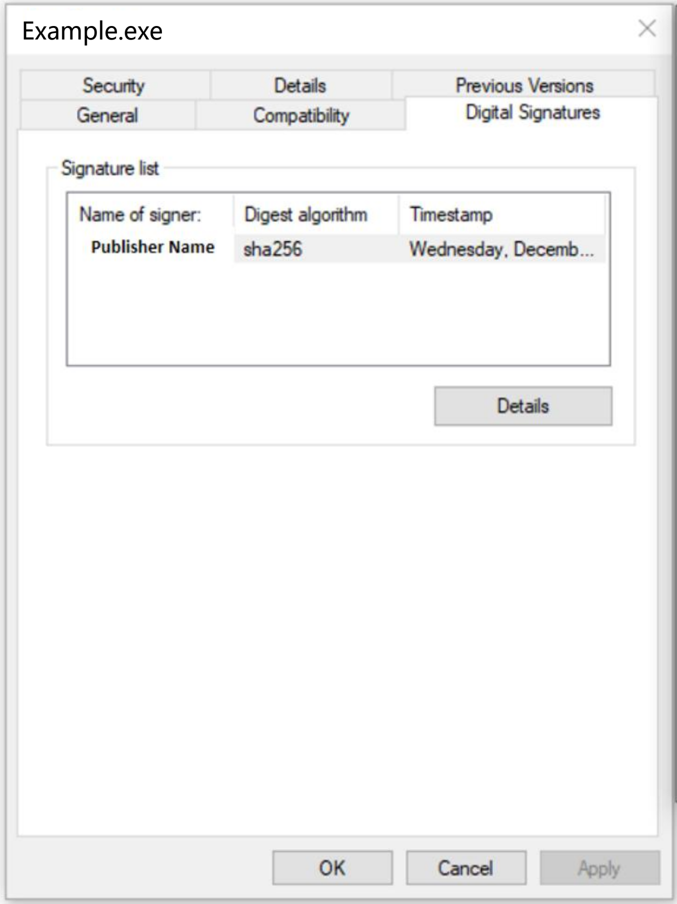
Global Software Supply-Chain Attack Statistics (2019–2024)*

* *https://www.sonatype.com/hubfs/SSCR-2024/SSCR_2024-FINAL-10-10-24.pdf*

Code Signing is a Vital Mechanism for Protecting Software

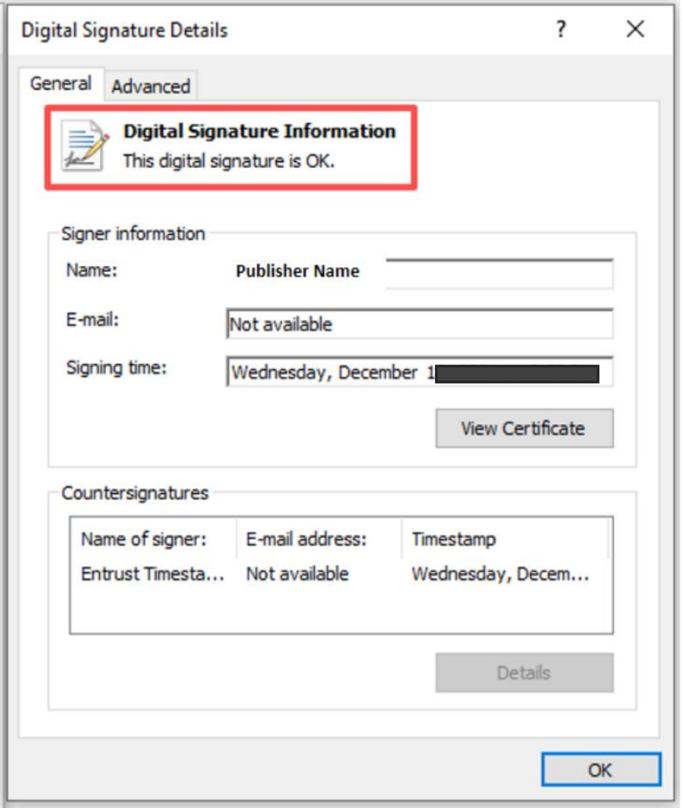
- **Security goal:** Code signing ensures software authenticity and integrity
- Developers obtain CA-issued certificates to digitally sign software





Signature list

Name of signer:	Digest algorithm	Timestamp
Publisher Name	sha256	Wednesday, Decemb...



Digital Signature Information
This digital signature is OK.

Signer information

Name: **Publisher Name**

E-mail: Not available

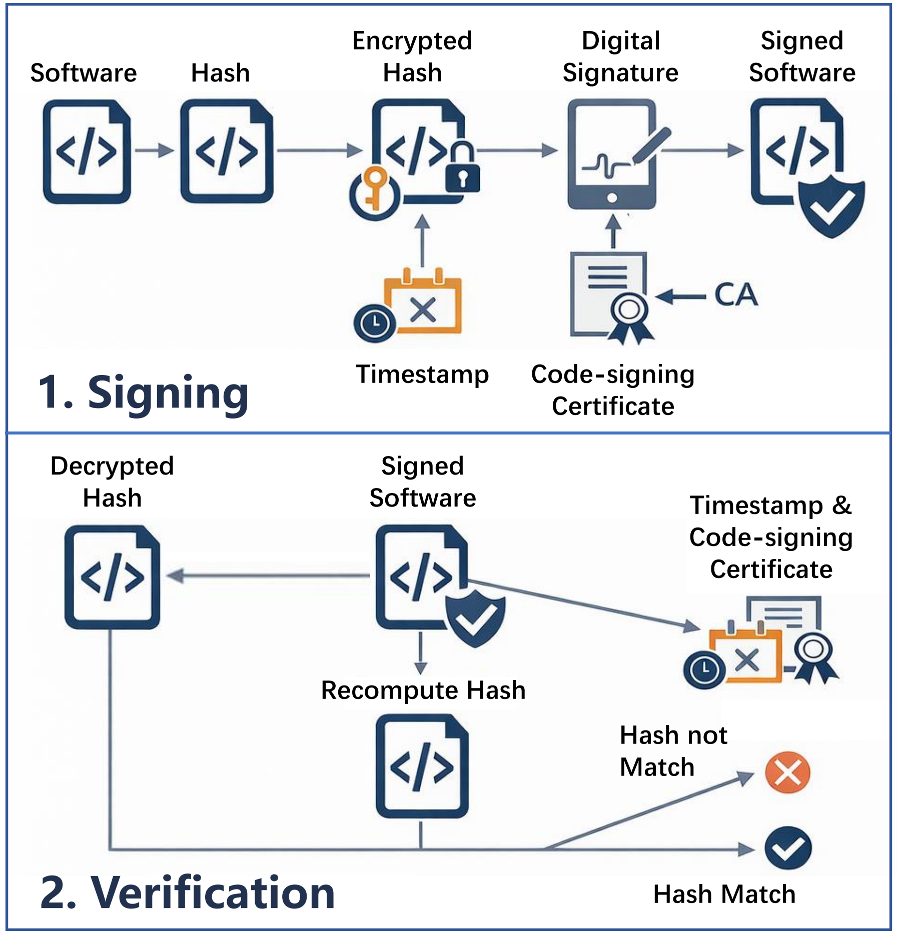
Signing time: Wednesday, December 1[redacted]

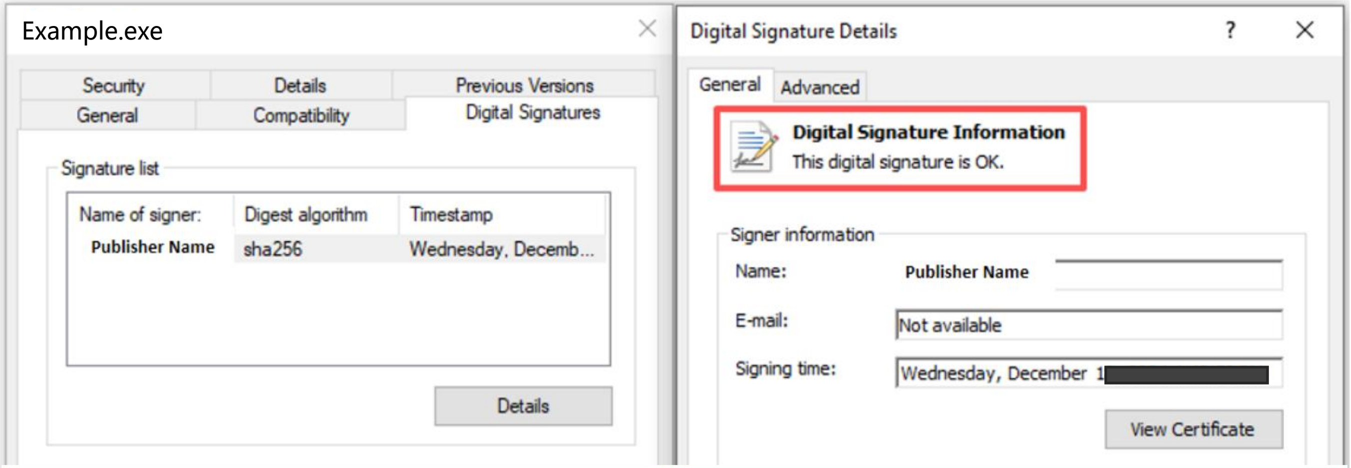
Countersignatures

Name of signer:	E-mail address:	Timestamp
Entrust Timesta...	Not available	Wednesday, Decem...

Code Signing is a Vital Mechanism for Protecting Software

- **Security goal:** Code signing ensures software authenticity and integrity
- Developers obtain CA-issued certificates to digitally sign software





Signature list

Name of signer:	Digest algorithm	Timestamp
Publisher Name	sha256	Wednesday, Decemb...

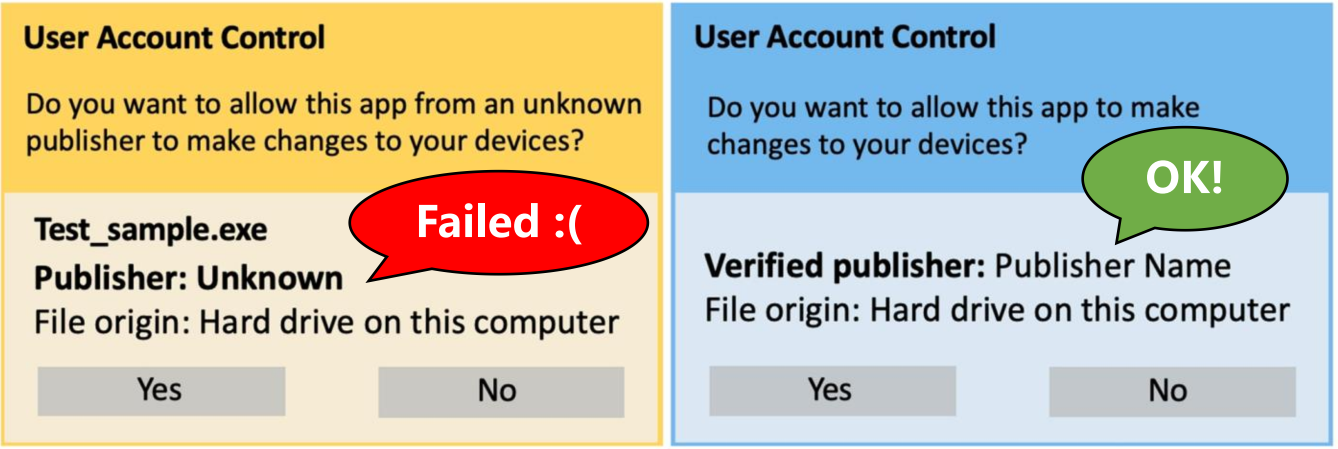
Digital Signature Information
This digital signature is OK.

Signer information

Name: Publisher Name

E-mail: Not available

Signing time: Wednesday, December 1[redacted]



User Account Control

Do you want to allow this app from an unknown publisher to make changes to your devices?

Test_sample.exe

Publisher: Unknown

File origin: Hard drive on this computer

User Account Control

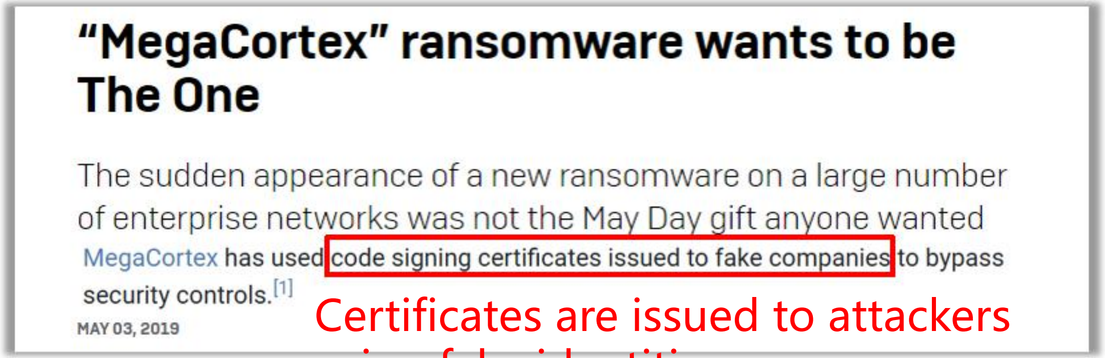
Do you want to allow this app to make changes to your devices?

Verified publisher: Publisher Name

File origin: Hard drive on this computer

However, Code Signing is a Double-edged Sword

- Designed for security, code signing can be exploited to undermine trust
- **Code-signing abuse:** attackers leverage flaws in the code signing PKI to sign malware, bypassing checks by operating systems and antivirus software



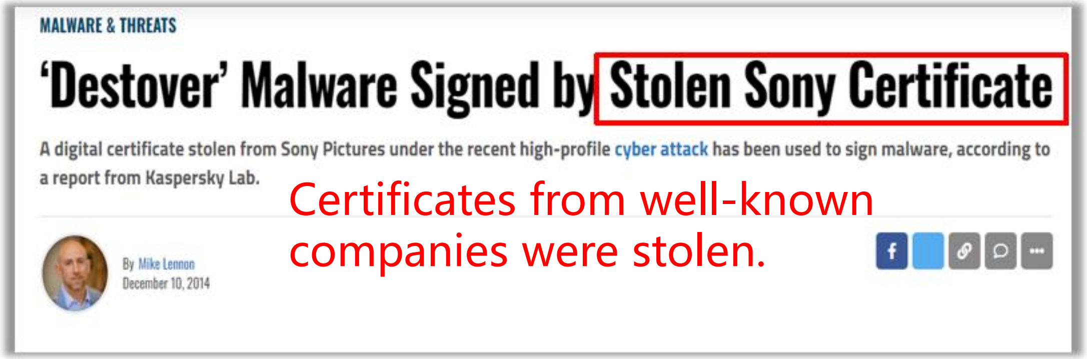
“MegaCortex” ransomware wants to be The One

The sudden appearance of a new ransomware on a large number of enterprise networks was not the May Day gift anyone wanted

MegaCortex has used code signing certificates issued to fake companies to bypass security controls.[1]

MAY 03, 2019

Certificates are issued to attackers using fake identities.



MALWARE & THREATS

‘Destover’ Malware Signed by Stolen Sony Certificate

A digital certificate stolen from Sony Pictures under the recent high-profile cyber attack has been used to sign malware, according to a report from Kaspersky Lab.

By Mike Lennon

December 10, 2014

Certificates from well-known companies were stolen.

- Code-signing security incidents keep emerging
  - Since 2021, VirusTotal found ~1M malware samples with abused signatures*
  - Code-signing abuse has become a common tactic in APT (e.g., Stuxnet)

* *https://blog.virustotal.com/2022/08/deception-at-scale.html*

Research Questions and Challenges

- **Research Goal:** Identify code-signing PKI flaws, propose effective mitigations

Q1: What is the status of the code-signing abuse ecosystem?

Q2: What flaws do abusers exploit, and what strategies do they employ?

Q3: What are the root causes of code-signing abuse and how to mitigate it?

Research Questions and Challenges

- **Research Goal:** Identify code-signing PKI flaws, propose effective mitigations

Q1: What is the status of the code-signing abuse ecosystem?

→ **Challenge-1:** Closed code-signing ecosystem hinders access to large-scale malware datasets

Q2: What flaws do abusers exploit, and what strategies do they employ?

→ **Challenge-2:** Lacking ground truth hinders identifying and classifying abuse

Q3: What are the root causes of code-signing abuse and how to mitigate it?

→ **Challenge-3:** Opaque CA operations hinder root-cause analysis

Our Work

Identifying Code Signing Abuse Methodology

- Developed a new fine-grained classification method
- Built the largest labeled dataset with 43,286 abused certificates

Understanding the Code Signing Abuse Ecosystem

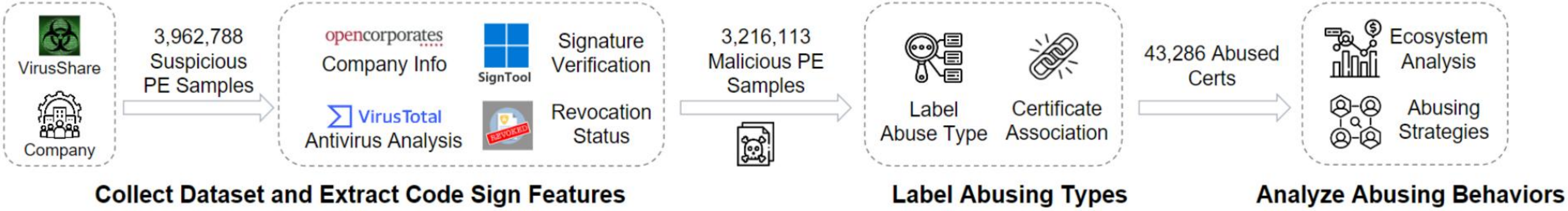
- Abuse is widespread, affecting 46 CAs across 114 countries/regions
- Countermeasures are limited: revocation rate is only 17.56%
- First discovered 3,789 Ghost Certificates

Security analysis of Code Signing Abuse Strategies

- Discovered 5 types of abuse strategies
- Found 59.12% of abused certificates are polymorphic
- Conducted case studies on real-world evasion of checks

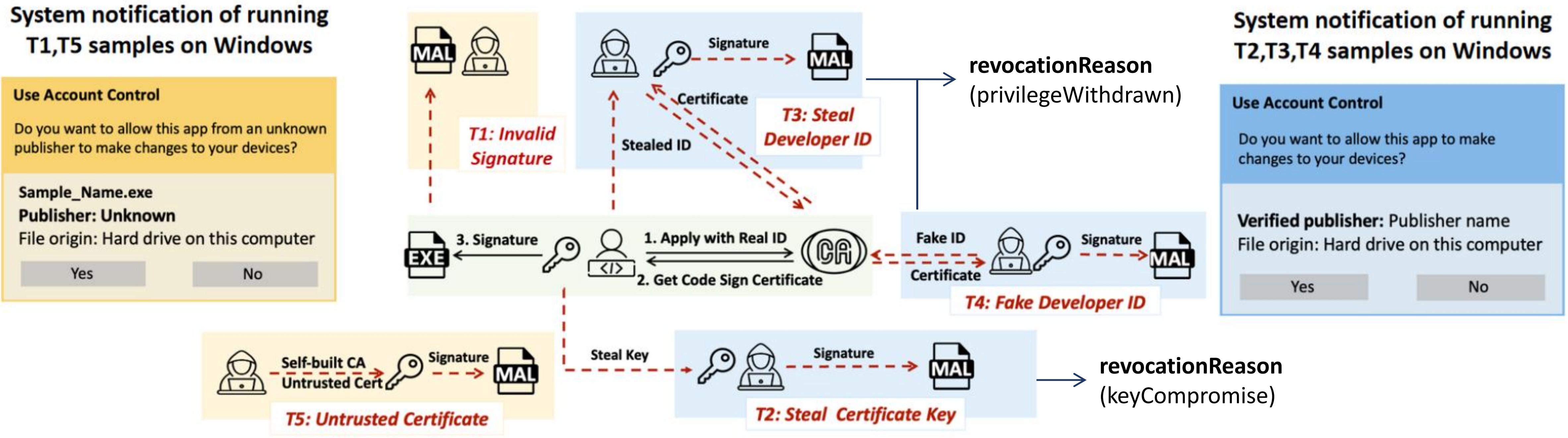
Overview of Data Processing Flow

- We built the **largest code-signing abuse dataset** to date, with **3,216,113** signed malicious samples and **43,286** abused certificates
- Our sample collection combined **private** (partner security company) and **public** datasets (VirusShare)
- For fine-grained classification and analysis, we added **extra features** (e.g., verification results, revocation status, and business registry data)



Methodology

- We categorize abuse into **five types** based on attackers' methods.
  - **Step-I:** Use SignTool to filter samples signed by *Invalid/untrusted certificates* (T1, T5)
  - **Step-II:** Use revocation reasons to separate *theft* (T2) and *impersonation* (T3, T4)
  - **Step-III:** Use business registries to separate *Stolen ID* (T3) and *fake ID* (T4)



System notification of running T1,T5 samples on Windows

Use Account Control

Do you want to allow this app from an unknown publisher to make changes to your devices?

Sample_Name.exe
Publisher: Unknown
File origin: Hard drive on this computer

Yes No

System notification of running T2,T3,T4 samples on Windows

Use Account Control

Do you want to allow this app to make changes to your devices?

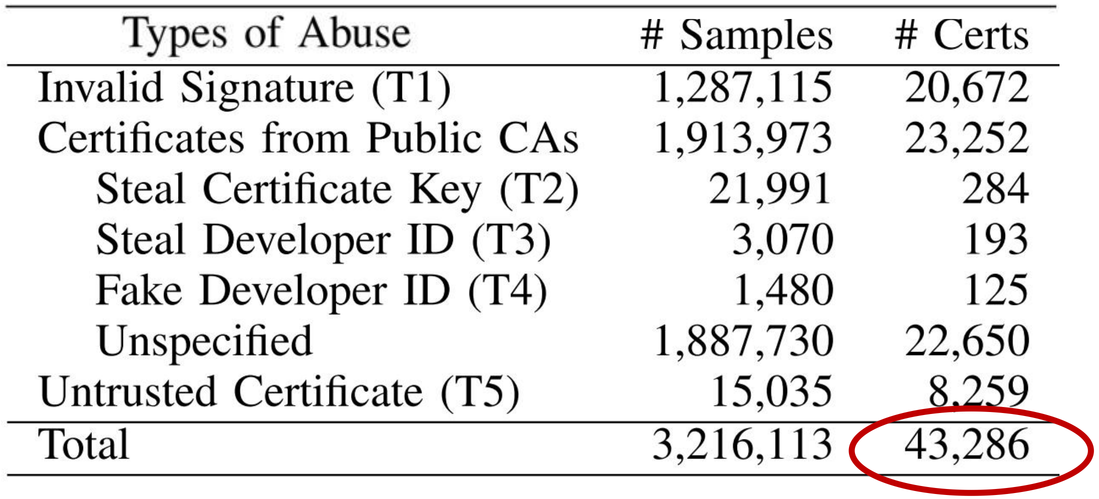
Verified publisher: Publisher name
File origin: Hard drive on this computer

Yes No

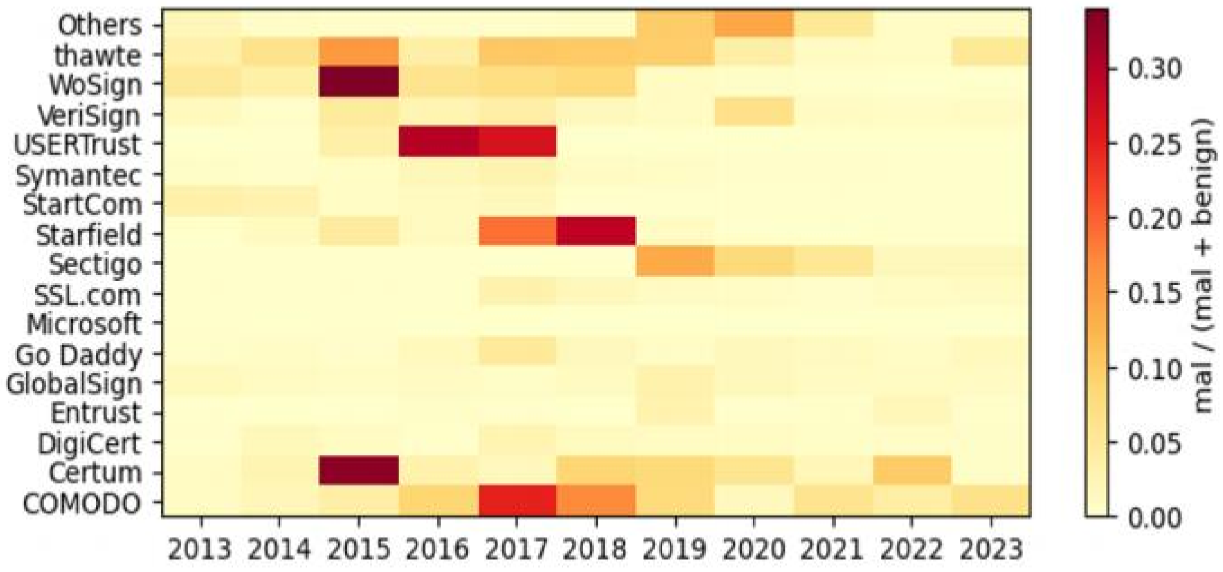
Finding 1: Code Signing Abuse Remains Widespread Globally

- Through fine-grained classification, we identified **43,286 abused certificates** and categorized them into five abuse types

- **Code signing abuse remains widespread**, affecting certificates from **114 countries** issued by **46 CAs**

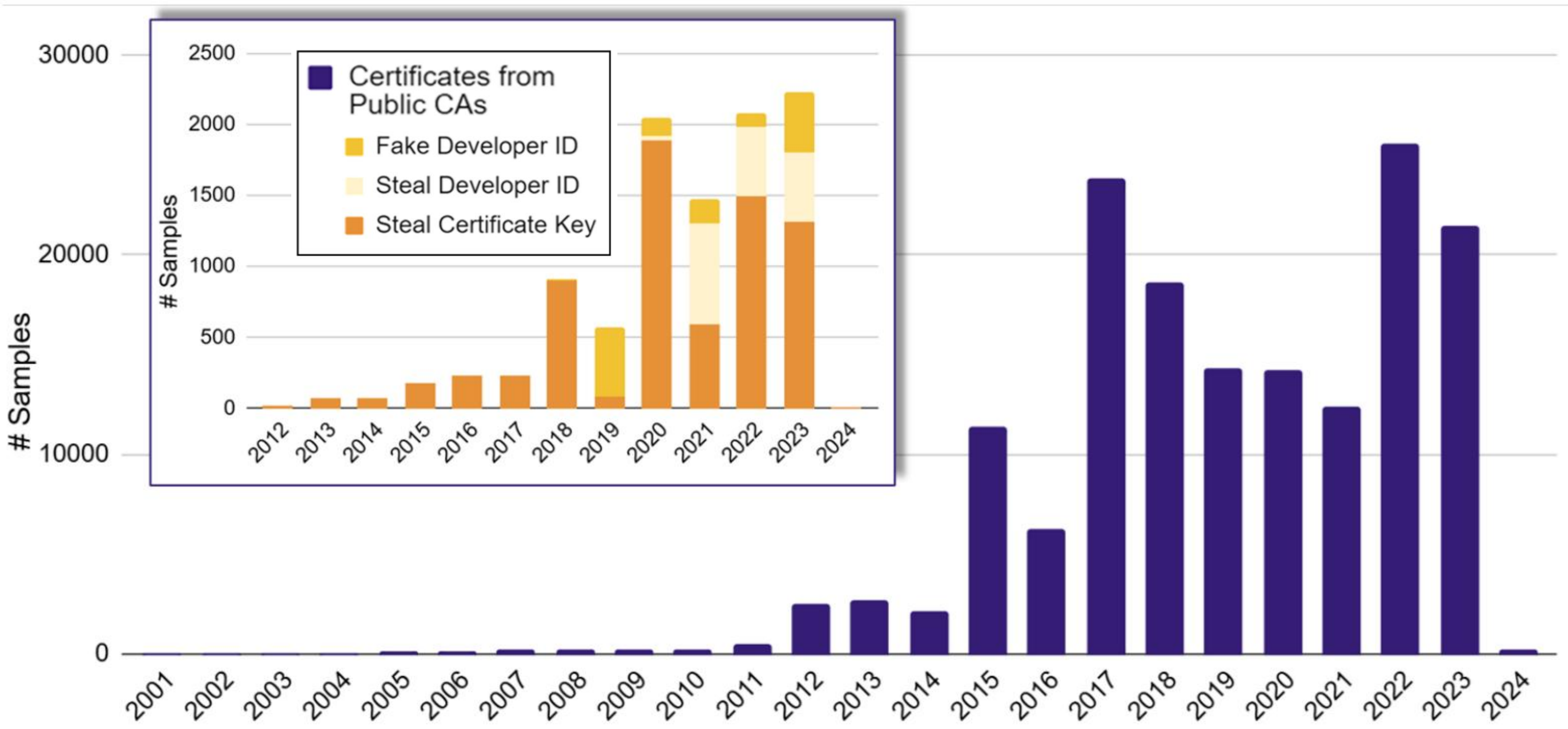


Types of Abuse	# Samples	# Certs
Invalid Signature (T1)	1,287,115	20,672
Certificates from Public CAs	1,913,973	23,252
Steal Certificate Key (T2)	21,991	284
Steal Developer ID (T3)	3,070	193
Fake Developer ID (T4)	1,480	125
Unspecified	1,887,730	22,650
Untrusted Certificate (T5)	15,035	8,259
Total	3,216,113	43,286



Finding 2: Abuse is Becoming More Advanced and Stealthy

- In recent years, **advanced abuse** (e.g., *stolen certificate keys, stolen Developer IDs, fake Developer IDs*) has grown sharply and is hard to detect



Advanced abuse samples rose **4×** in 2017–2023 versus 2010–2016

66.04d

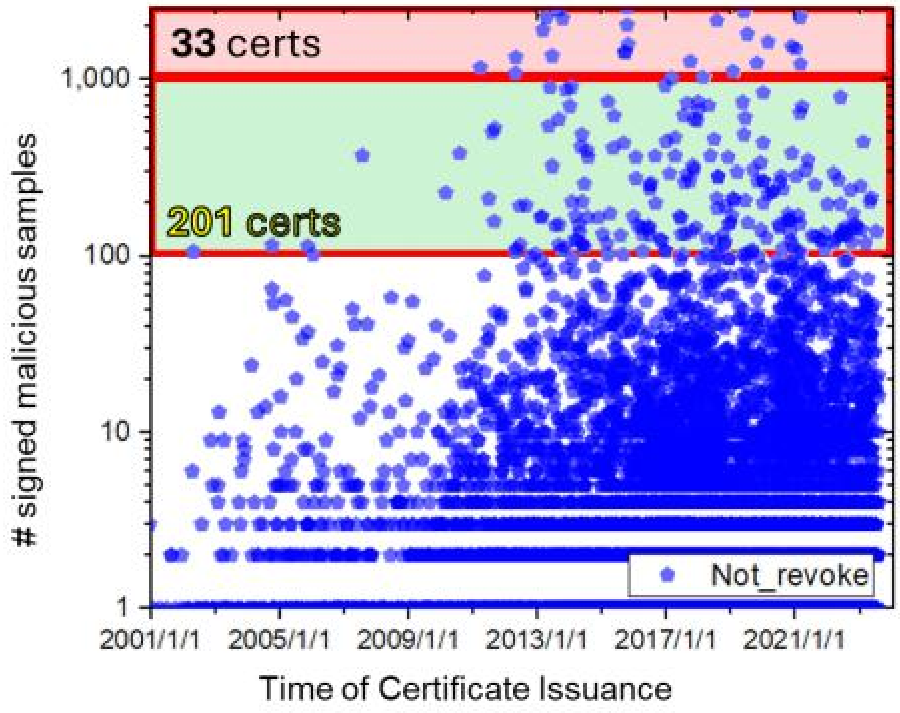
Average dwell time for malware signed by **stolen** certificates

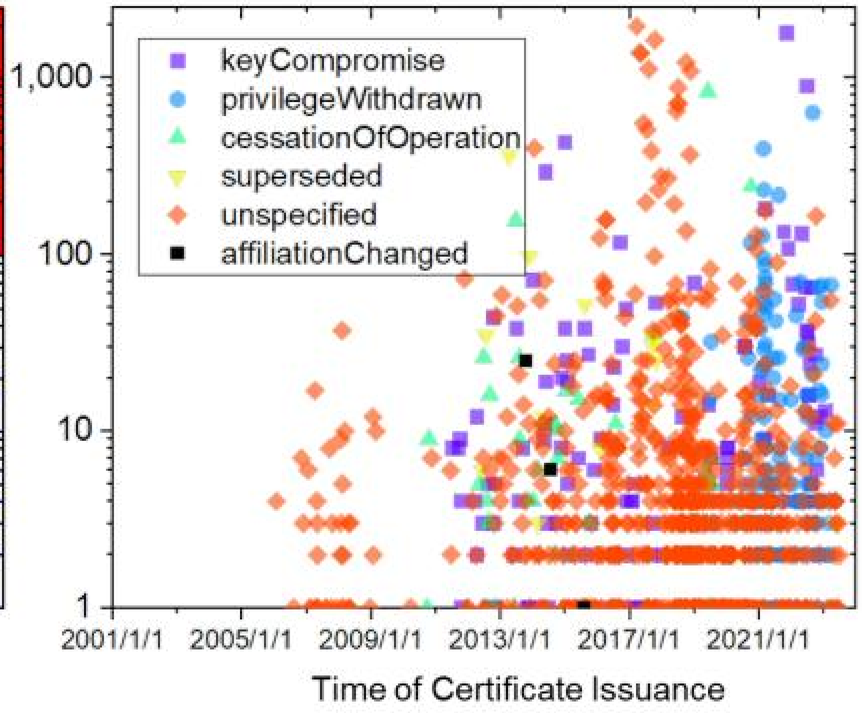
77.78%

of **stolen** certificates later signed benign-ware after malware

Finding 3: Low Revocation Rates Raise Serious Concerns

- The most effective way to block abused certificates is **revocation**, yet the revocation rate is only **17.56%**
- Although CAs provide increasingly **detailed revocation information** (e.g., reasonCode), **23.78%** of revocationDate remains inaccurate



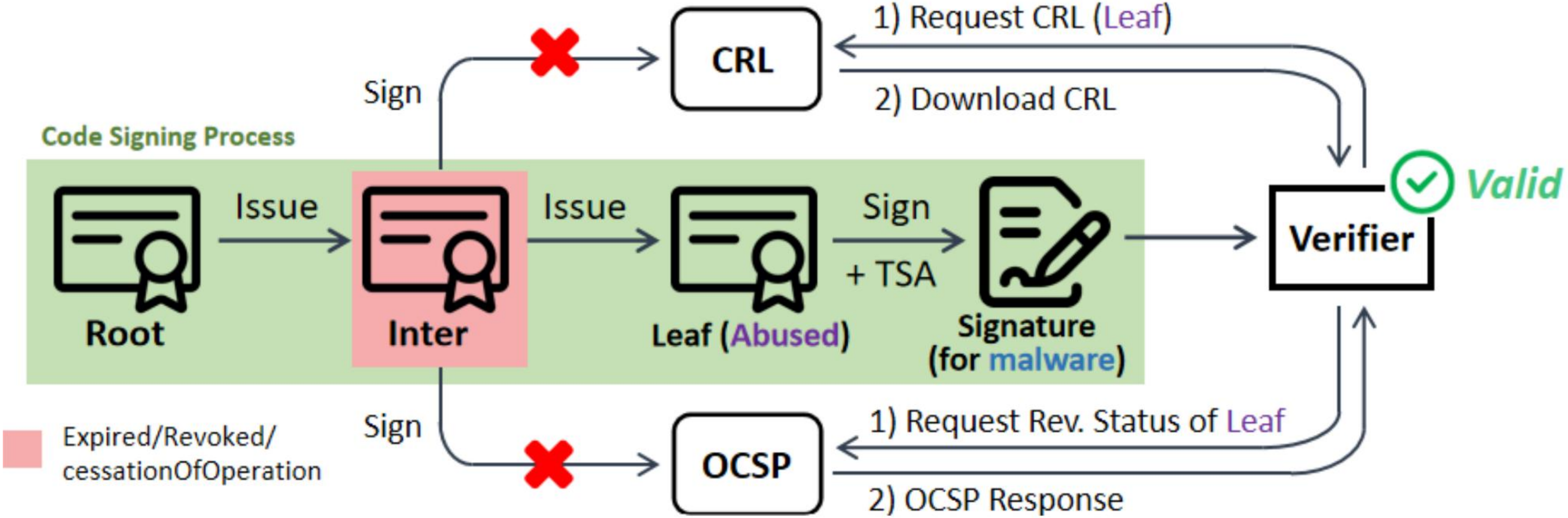


91,346

samples could still pass client-side validation

Finding 4: Ghost Certificates are a Hidden Bottleneck

- **Ghost certificates:** code signing certificates that have been abused but cannot be revoked due to design flaws
- At least **38.96%** of unrevoked abused certificates qualify as ***ghost certificates***



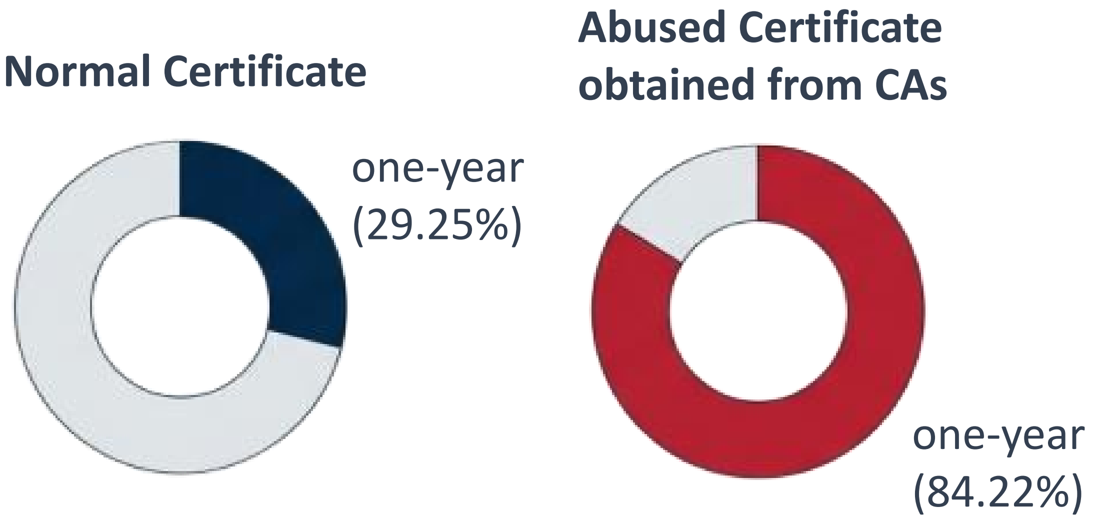
Strategy 1~2: Crafted Tactics during Certificate Issuance

- **Strategy 1:** Exploiting differences in identity-verification strictness across countries

Rank	Benign Cert	Fake ID Cert
1	United States	Russia
2	China	Armenia (85th)
3	Germany	Vietnam (48th)

Insight: Some countries are favored by identity forgers, suggesting inconsistent CA ID-check rigor across countries

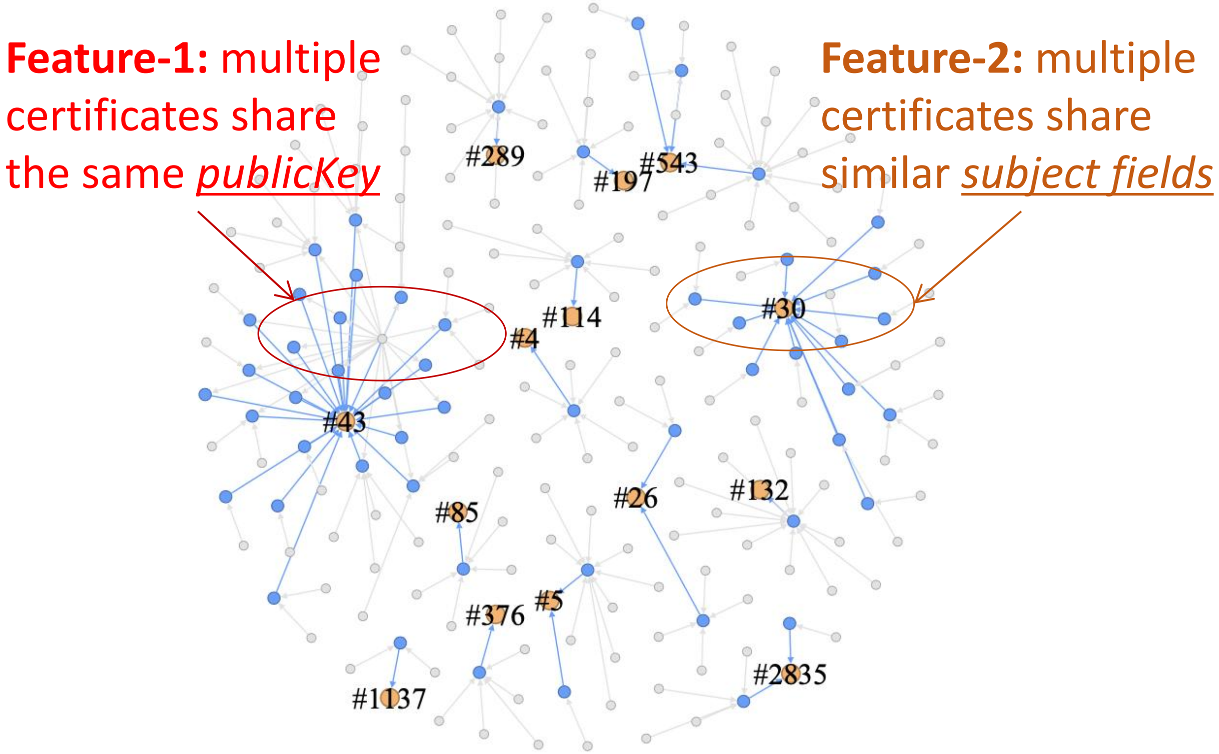
- **Strategy 2:** Using short-lived certificates to reduce cost and risk



Insight: Short-lived certificates are cheaper (typically ~$500), and revocation causes smaller losses for abusers

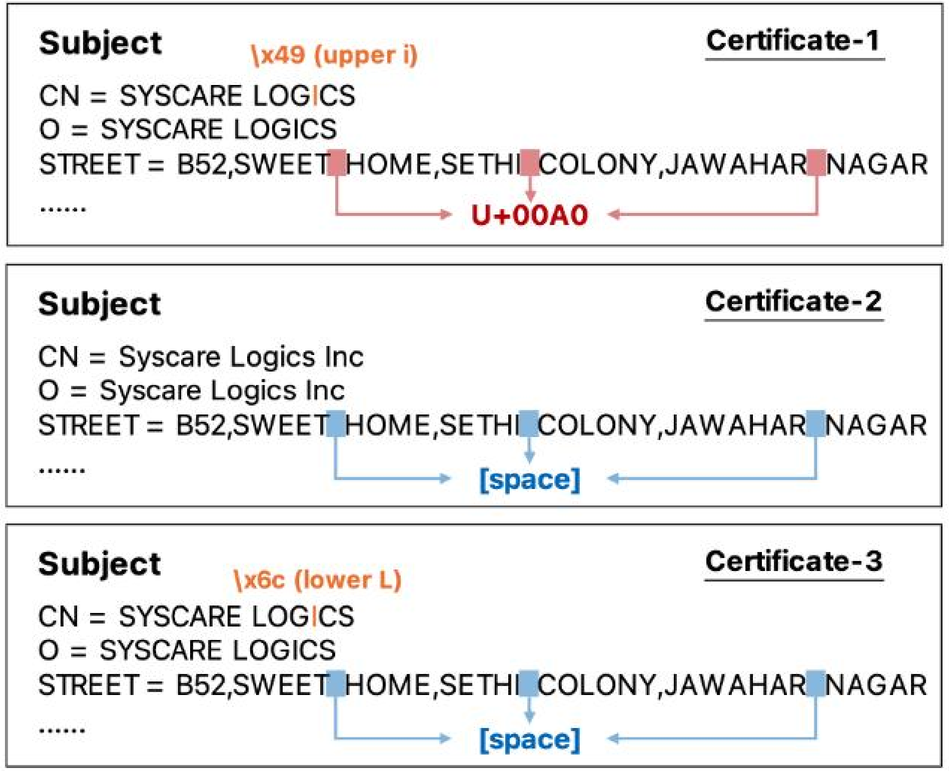
Strategy 3: Exploit Certificate Polymorphism to Evade Checks

- **Certificate polymorphism:** the same identity entity obtains multiple certificates from the same or different CAs using the same (or slightly modified) identity.
- **Security Impact:** helping abusers reduce costs and evade detection



Feature-1: multiple certificates share the same *publicKey*

Feature-2: multiple certificates share similar *subject fields*



Subject \x49 (upper i) **Certificate-1**

CN = SYSCARE LOGICS
O = SYSCARE LOGICS
STREET = B52,SWEET HOME,SETHI COLONY,JAWAHAR NAGAR
......

U+00A0

Subject **Certificate-2**

CN = Syscare Logics Inc
O = Syscare Logics Inc
STREET = B52,SWEET HOME,SETHI COLONY,JAWAHAR NAGAR
......

[space]

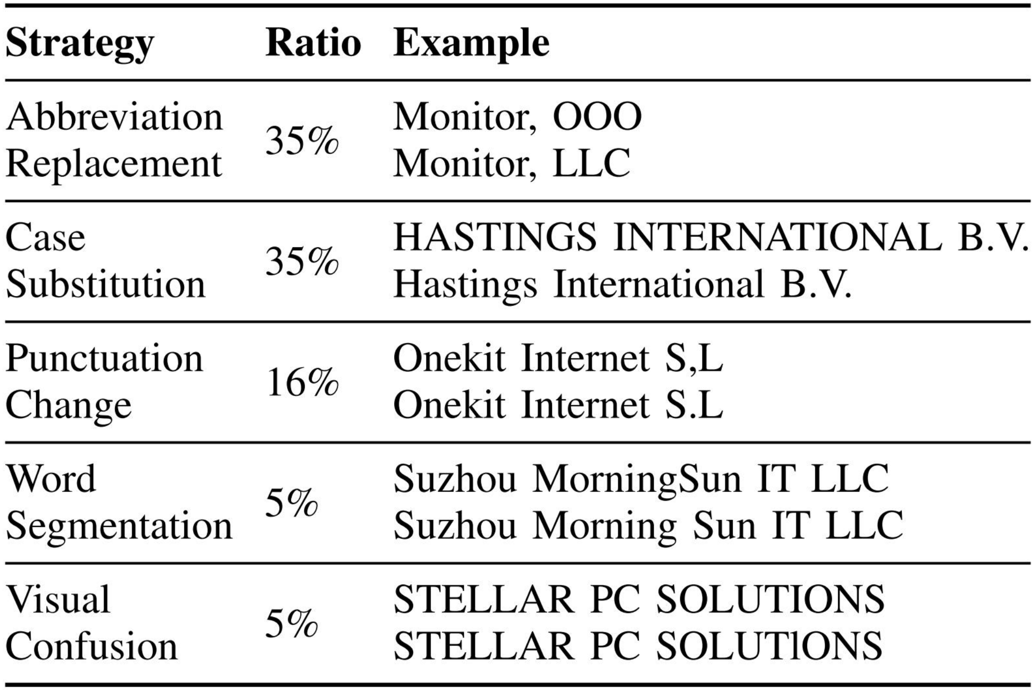
Subject \x6c (lower L) **Certificate-3**

CN = SYSCARE LOGlCS
O = SYSCARE LOGICS
STREET = B52,SWEET HOME,SETHI COLONY,JAWAHAR NAGAR
......

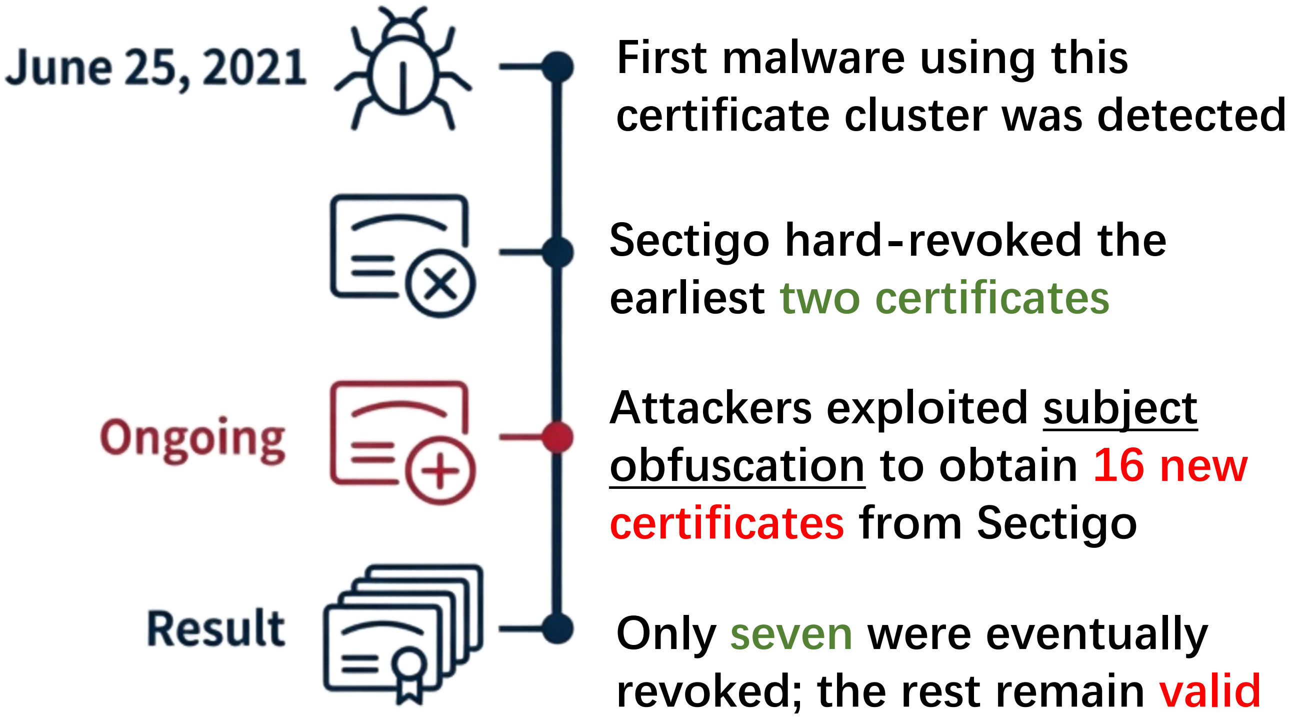
[space]

Strategy 3: Exploit Certificate Polymorphism to Evade Checks

- We identified **3,484** polymorphic certificate clusters containing **13,747 (59.12%)** abused certificates, which exhibit diverse strategies
- Among them, **315 (9.04%)** clusters show inconsistent revocation



Strategy	Ratio	Example
Abbreviation Replacement	35%	Monitor, OOO Monitor, LLC
Case Substitution	35%	HASTINGS INTERNATIONAL B.V. Hastings International B.V.
Punctuation Change	16%	Onekit Internet S,L Onekit Internet S.L
Word Segmentation	5%	Suzhou MorningSun IT LLC Suzhou Morning Sun IT LLC
Visual Confusion	5%	STELLAR PC SOLUTIONS STELLAR PC SOLUTlONS



June 25, 2021 — First malware using this certificate cluster was detected

Sectigo hard-revoked the earliest two certificates

Ongoing — Attackers exploited subject obfuscation to obtain 16 new certificates from Sectigo

Result — Only seven were eventually revoked; the rest remain valid

Root Cause - Weakness of Code Signing PKI

Weak governance on the CA side

Lack of rigor and standardization during certificate issuance

- Loose identity verification
- No strict constraints on subject

Unproactive abuse governance

- Reliance on passive reports
- Failure to ban high-risk entities

Design flaws in code signing PKI

“Ghost certificate” issue

- Existing revocation mechanisms are limited
- Broken revocation infrastructure is overlooked

Single point of client reliance

- Windows client verification fully relies on CRL/OCSP
- Lacks robust fallback checks

Recommendations for Code Signing PKI

Suggestions for CAs

1. Increase transparency of revocation and issuance

- Disclose high-risk entities
- Build transparency logs

2. Proactively detect abuse

- Use antivirus engines to monitor malicious signing activities
- Audit polymorphic certificates

3. Establish standards for certificate subject names

Suggestions for Operating System

Mitigate “ghost certificates”

- Decouple CRL/OCSP checks from the code-signing chain
- Permit independent key rotation for the revocation infrastructure

Suggestions for security system

Adopt proactive abuse governance

- Aggregate threat intelligence to maintain blocklists of high-risk certificates

Open Science - Code Signing Abuse Dataset

- **Repository:** https://github.com/XingTuLab/Code_Signing_Abuse_Dataset
- **What's included:**
  - **CSV tables:** structured metadata for each abused certificate (e.g., hash, serial number, subject, issuer, validity period, abuse category), plus VirusTotal report links for one representative sample per certificate
  - **Certificate bundle (ZIP):** raw .cer files for all certificates (filename = certificate MD5)
- **Ethics considerations:**
  - Release malware-related abused certificates
  - Publish only confirmed revoked certificates
  - Provide VirusTotal URLs, no original samples

Tsinghua University

QI-ANXIN

ZGC LAB

Understanding the Status and Strategies of the Code Signing Abuse Ecosystem

Hanqing Zhao, **Yiming Zhang**, Lingyun Ying, Mingming Zhang,

Baojun Liu, Haixin Duan, Zi-Quan You, Shuhao Zhang

Email: zhaohq23@mails.tsinghua.edu.cn

https://github.com/XingTuLab/Code_Signing_Abuse_Dataset

NDSS SYMPOSIUM