The Broken Shield: Measuring Revocation Effectiveness in the Windows Code-Signing PKI

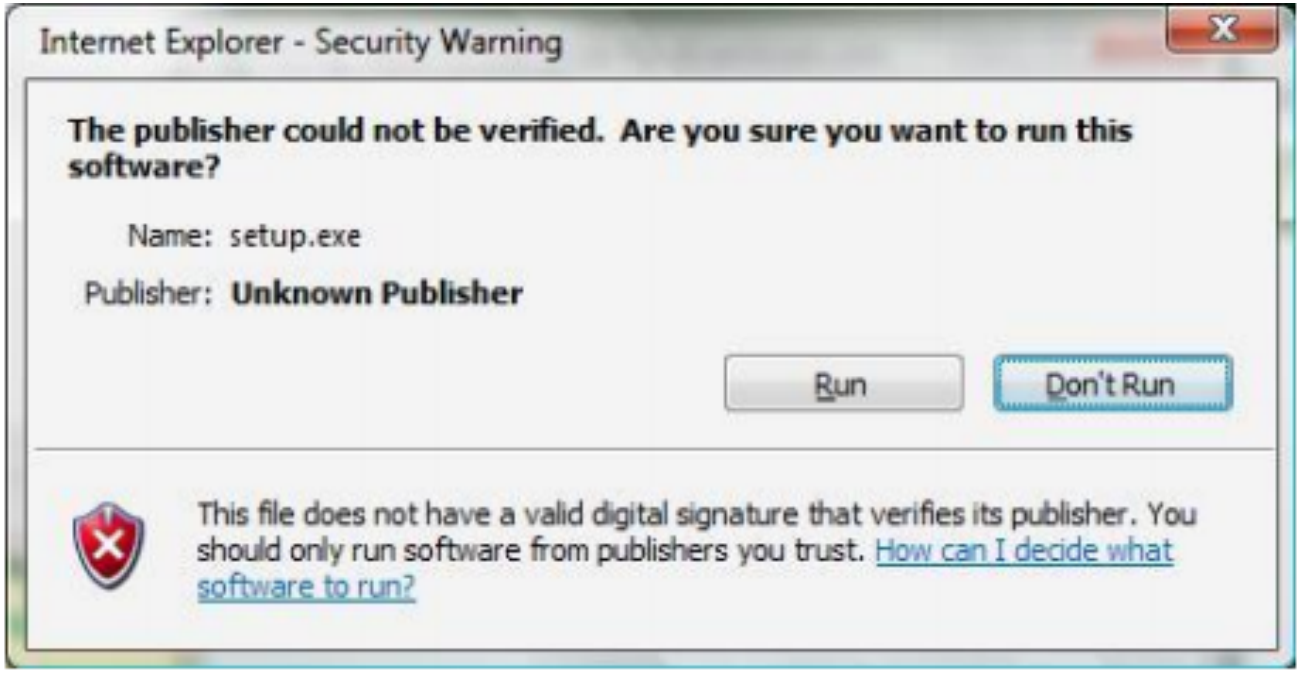
Doowon Kim[1], Bum Jun Kwon[1], Kristián Kozák[2],

Christopher Gates[3], and Tudor Dumitraș[1]

[1]University of Maryland, College Park, [2]Masaryk University, [3]Symantec

Why is the Code Signing PKI required?

- Nature of software distributed over the Internet
  - Unidentifiable software authors (publishers)
  - May be tampered



Internet Explorer - Security Warning

The publisher could not be verified. Are you sure you want to run this software?

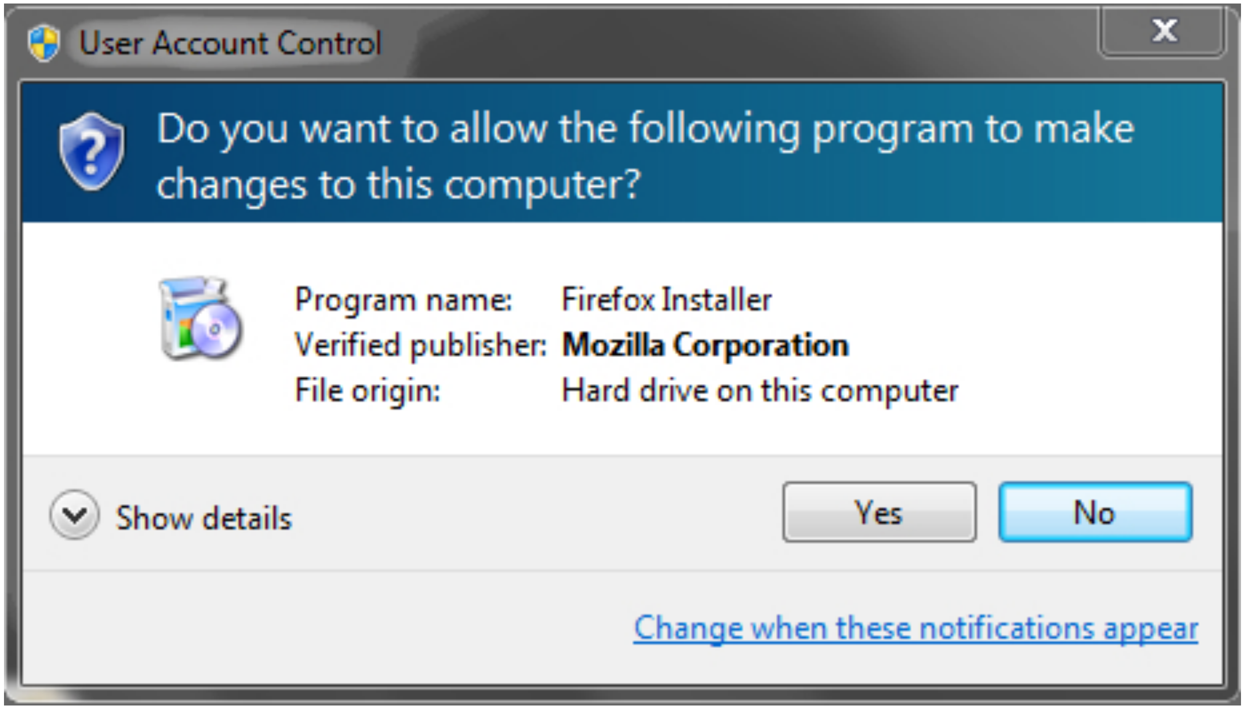
Name: setup.exe

Publisher: **Unknown Publisher**

This file does not have a valid digital signature that verifies its publisher. You should only run software from publishers you trust. How can I decide what software to run?

Why is the Code Signing PKI required?

- Code signing PKI helps establish …
  - **Authenticity** of publisher
  - **Integrity** of software



Do you want to allow the following program to make changes to this computer?

Program name: Firefox Installer
Verified publisher: **Mozilla Corporation**
File origin: Hard drive on this computer

Abuse and Primary Defense

- Abuse cases
  - Stuxnet
  - Black Market [1]
  - Etc.

- Primary defense: **Revocation**
  - Compromised certificates must be revoked
  - To make them no longer valid

1. Kozák et al. Issued for Abuse: Measuring the Underground Trade in Code Signing Certificate, WEIS 2018.

Motivation

- In our prior work, we found that 2/3 compromised certificates are not revoked [1]
- Why are the most not revoked yet?
- Furthermore, do CAs properly understand the code signing PKI and revoke compromised certificates without any mistakes?

We measure the effectiveness of revocations

[1] Kim et al. Certified Malware: Measuring Breaches of Trust in the Windows Code-Signing PKI, CCS 2017.

How to Revoke Potentially Compromised Certificates?

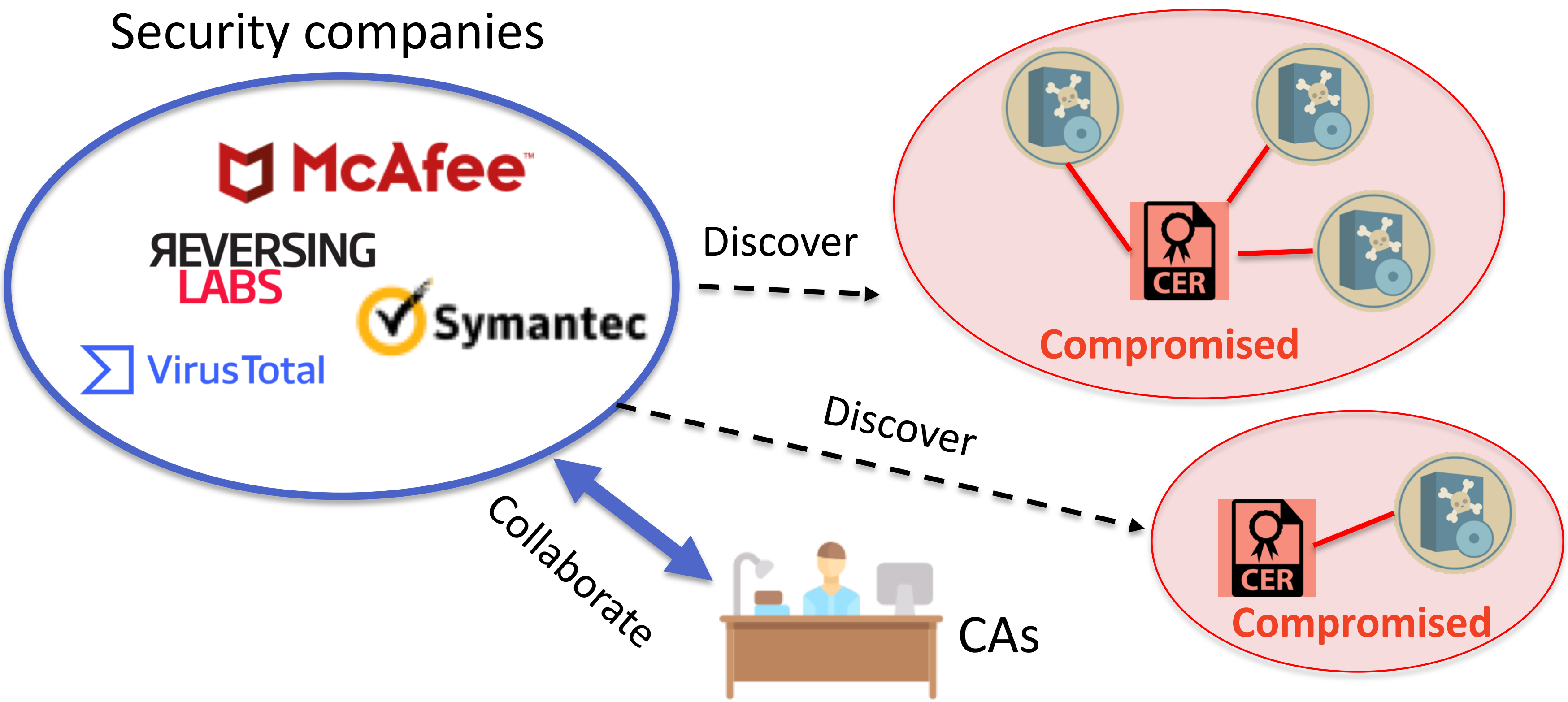
We identify **three steps** required:

1. Promptly discovery compromised certificates

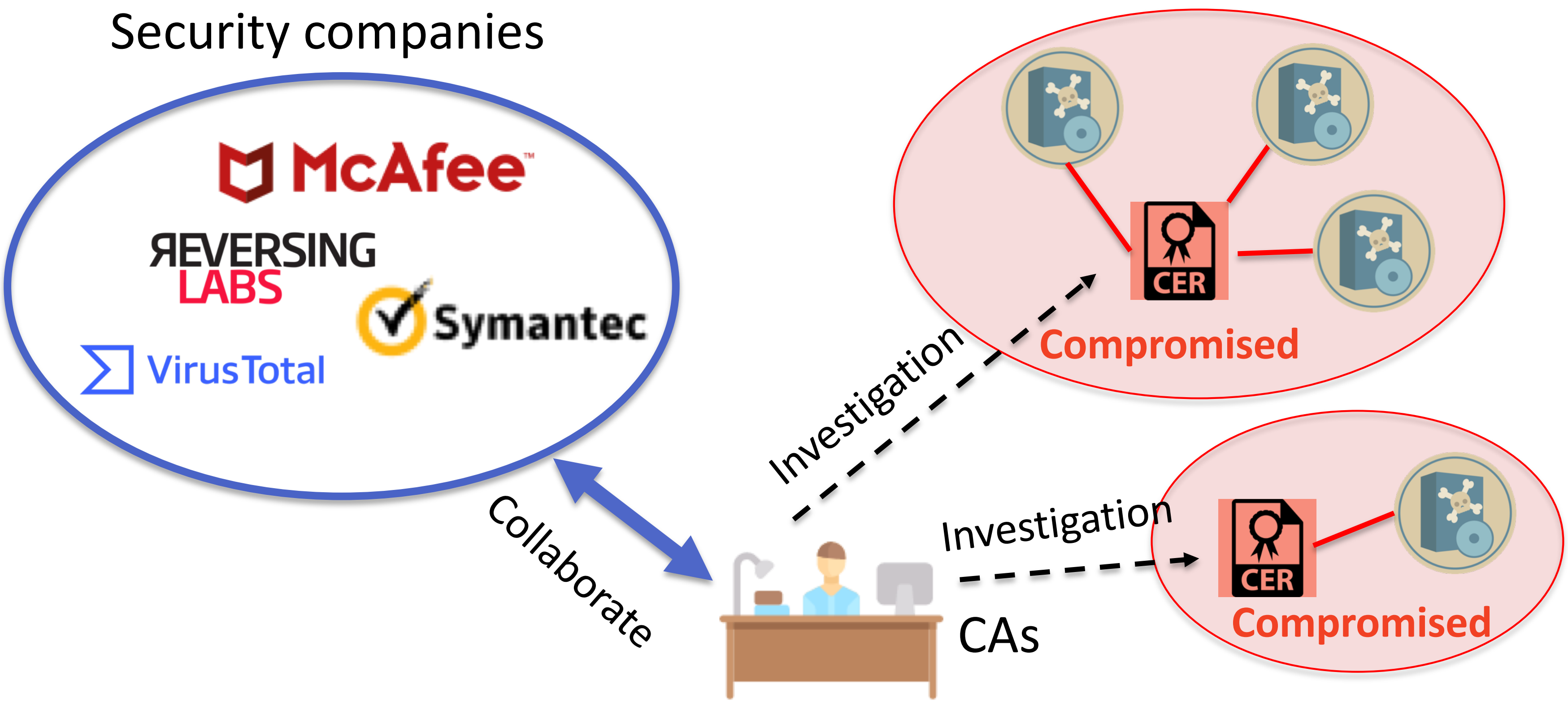
2. Invalidate all signed malware when revoking

3. Disseminate revocation information for clients

Step #1: Discover Compromised Certificates



Step #1: Discover Compromised Certificates

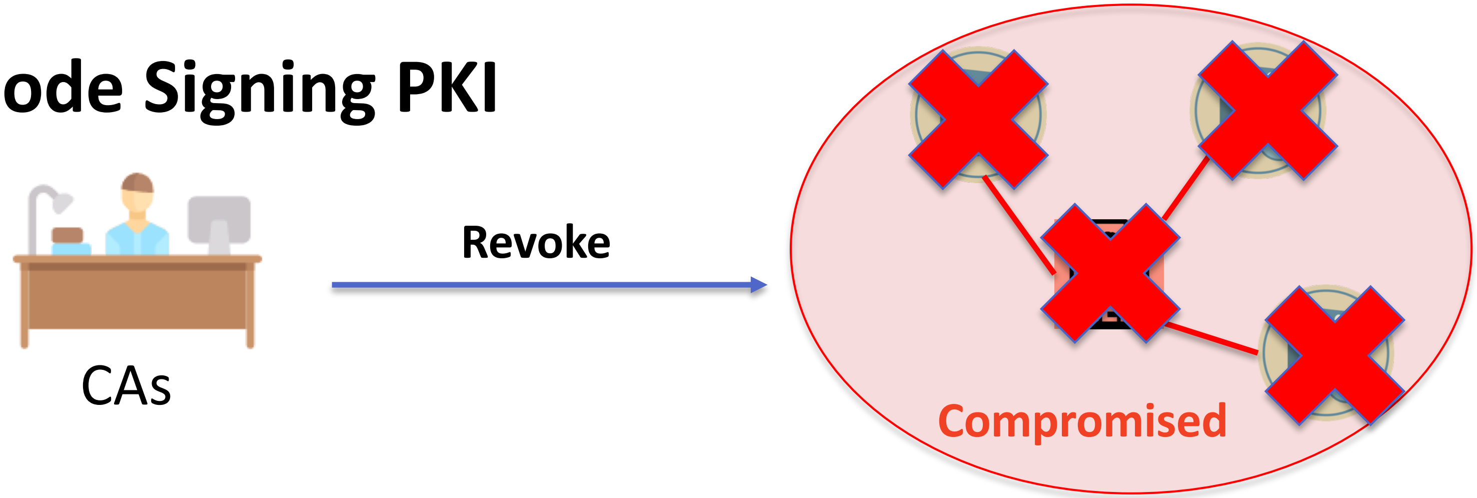


RQ1) How **promptly** do CAs discover and revoke compromised certificates after they appear in the wild?

→We found **delays of 5.6 months** to revoke compromised certificates

Step #2: Invalidate All Signed Malware

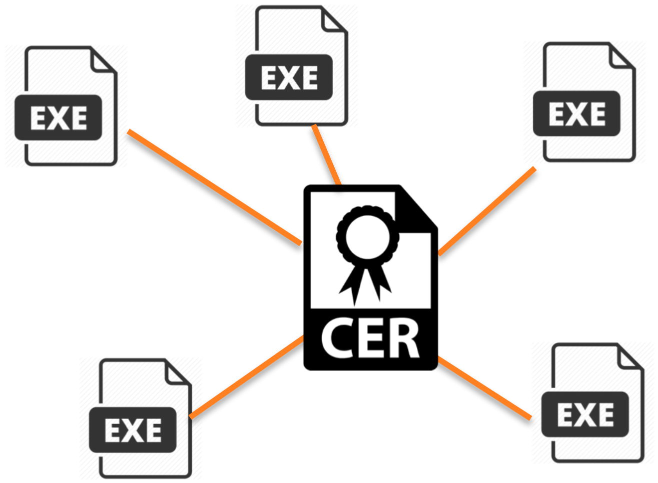
Code Signing PKI



Step #2: Invalidate All Signed Malware

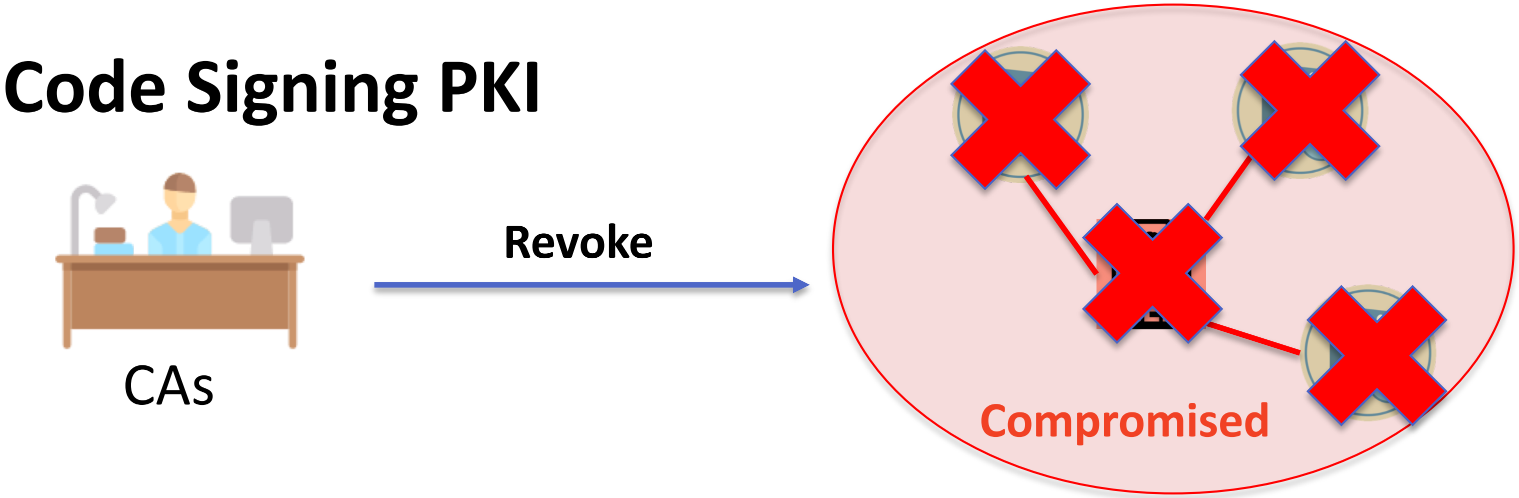
- One-to-many relationship
  - A certificate is used to sign numerous samples
  - C.f., TLS, one-to-one relationship

Code Signing PKI



TLS

Step #2: Invalidate All Signed Malware



Code Signing PKI

Revoke

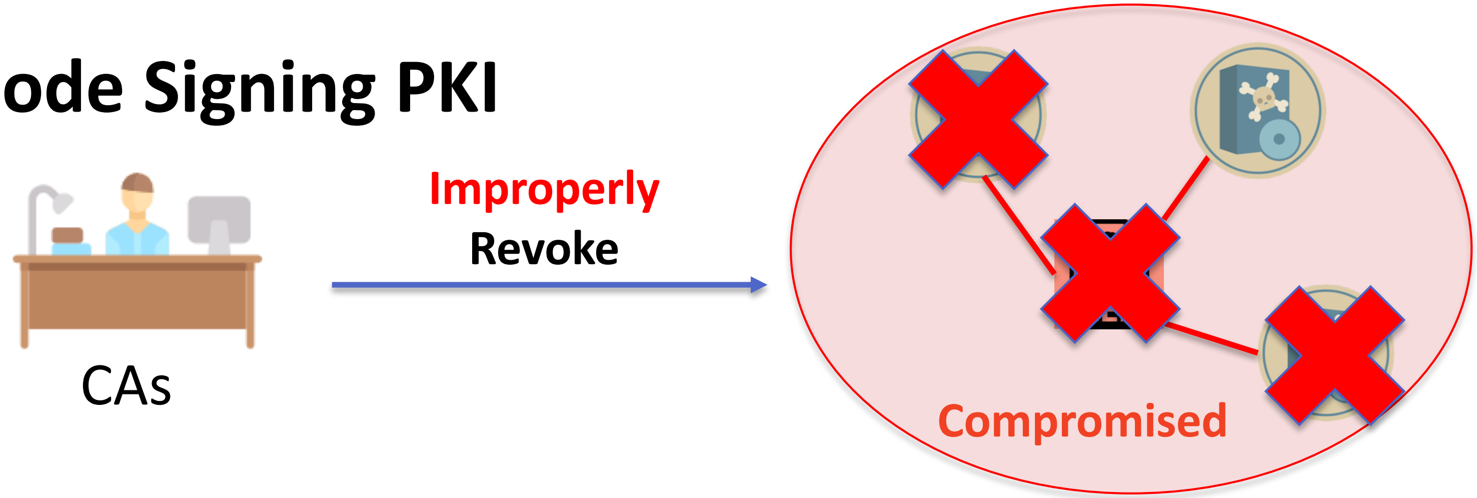
CAs

Compromised

RQ2) Do CAs properly revoke them and invalidate all malwares?

Step #2: Invalidate All Signed Malware

Code Signing PKI



CAs

Improperly Revoke

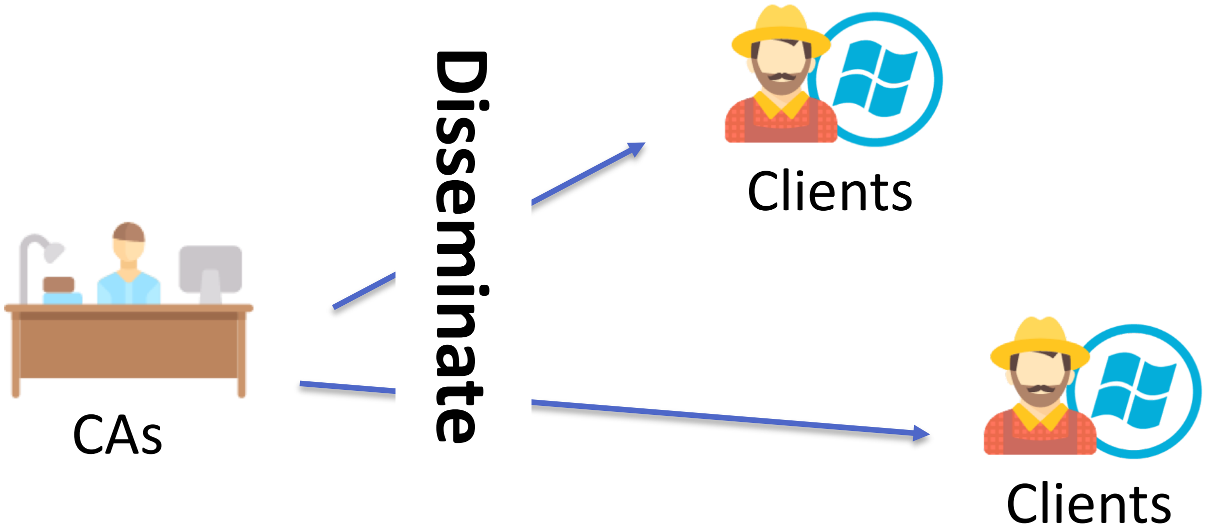
Compromised

RQ2) Do CAs properly revoke them and invalidate all malwares?

→ We found that CAs improperly revoke 5% compromised certificates and 5% signed malware are still valid

➔ **More critical and challenging than TLS**

Step #3: Disseminate Revocation Information



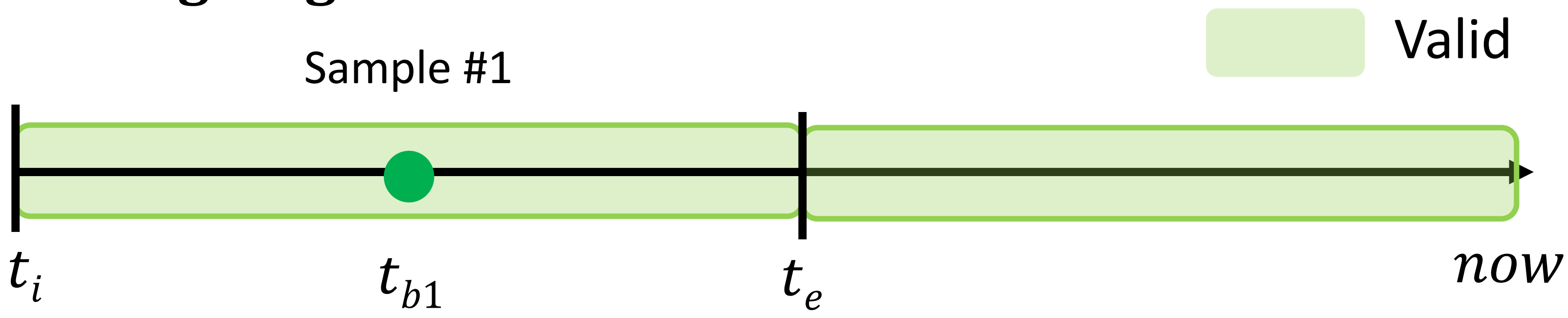
- Always-available for clients
- Must not remove expired certificates in CRLs

Trusted Timestamping

- Trusted creation timestamp of a program

- Extend trust in the program beyond expiration date

Code signing PKI



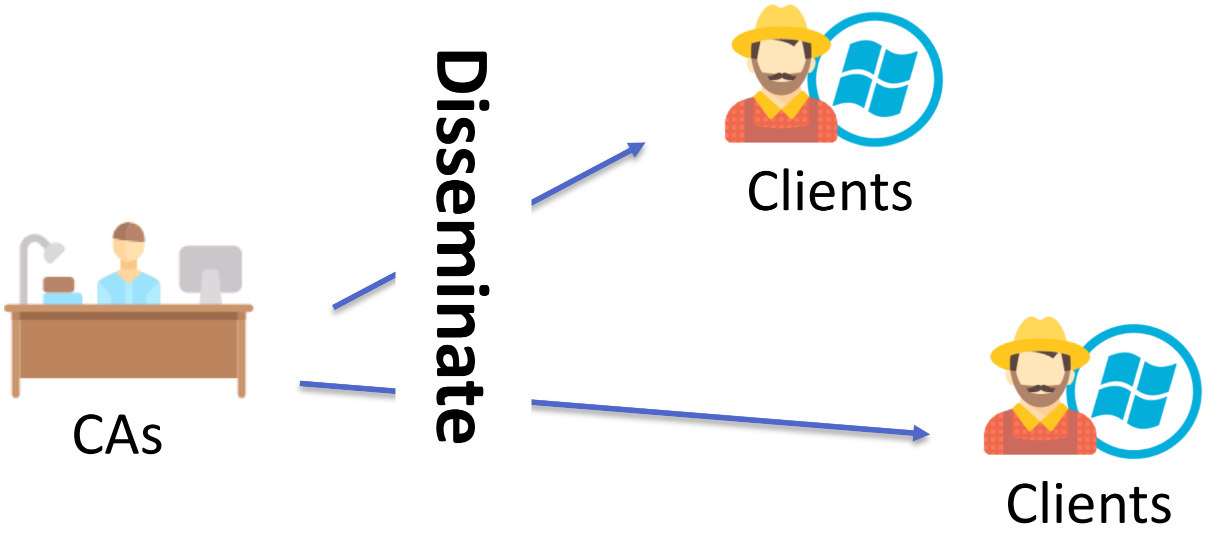
t_i: issue date

t_e: expiration date

Trusted Timestamping

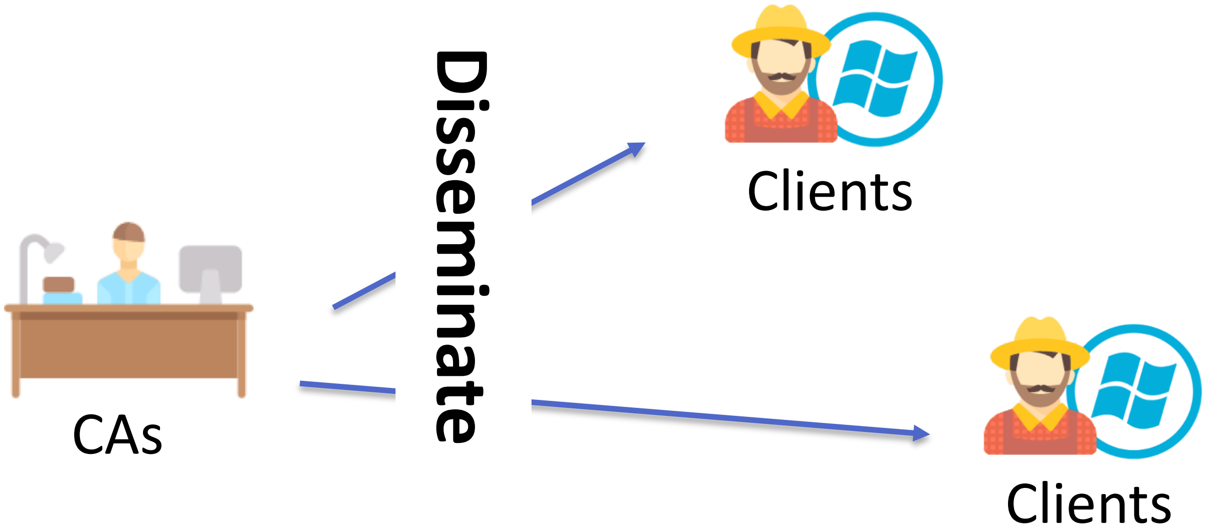
- Trusted creation timestamp of a program
- Extend trust in the program beyond expiration date
- Must care about even **expired certificates**

Step #3: Disseminate Revocation Information



RQ3) Do CAs properly maintain revocation information and disseminate it?

Step #3: Disseminate Revocation Information



RQ3) Do CAs properly maintain revocation information and disseminate it?

→We found that CAs removed 278 certificates from CRLs and improperly maintain infrastructures

➔**More critical** and **more challenging** than TLS

Contributions

- We identified the effective revocation process
  1. Discover compromised certificates
  2. Invalidate all signed malware when revoking
  3. Properly disseminate revocation information

- We measured the effective revocation process and showed that revocation in the code signing PKI is **more critical** and **more challenging** than TLS

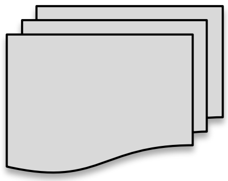
Outline

- **Data collection**
- Results: Effectiveness of revocation process
  - Discovery of compromised certificates
  - Invalidation of all signed malware
  - Dissemination of revocation information

Data Collection: Challenges

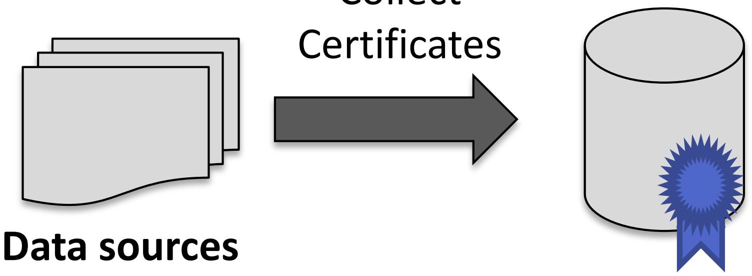
- No large corpus of code singing certificates
  - TLS: Censys.io, IPv4 scanning, Alexa 1M domains, etc
- Unable to know when certificates are revoked
  - Revocation date: The date that determines the validity of signed sample
  - C.f., TLS: The date at which the revocation took place

Data Collection



Data sources
(Symantec, MalSign, WINE…)

Data Collection

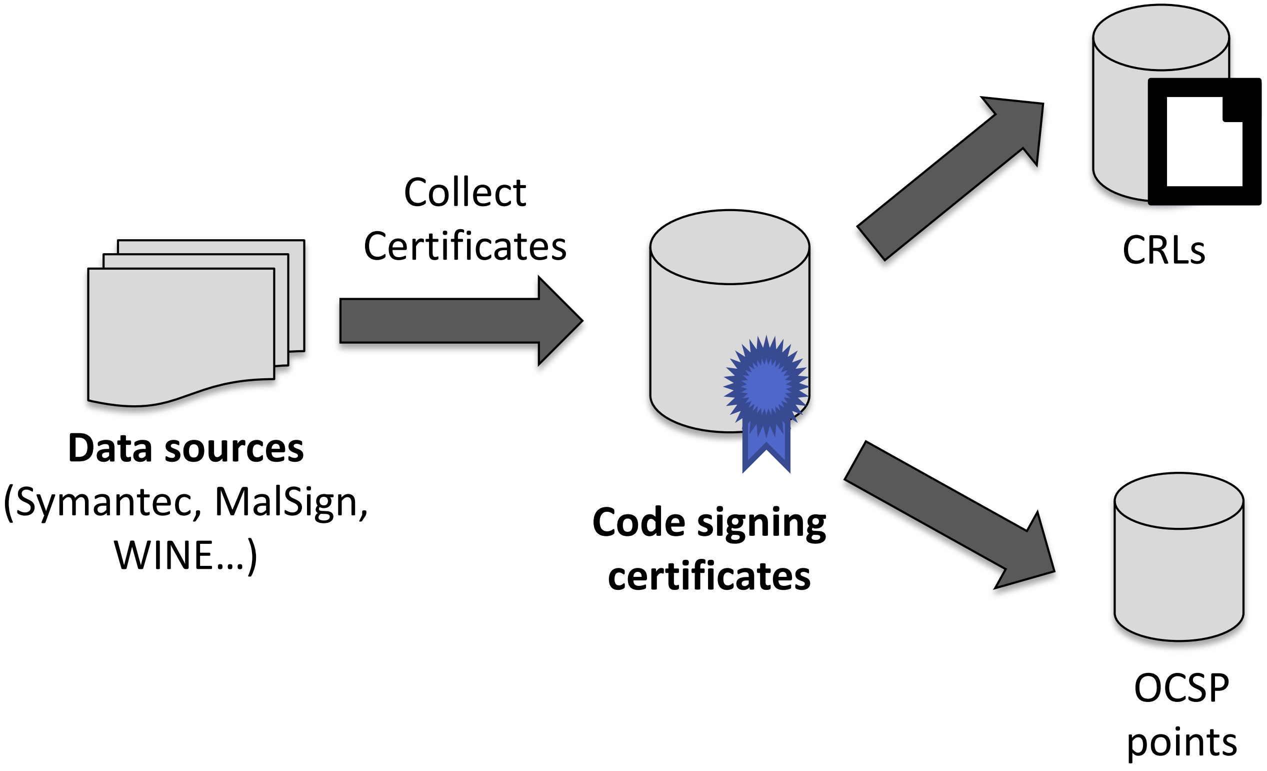


Collect Certificates

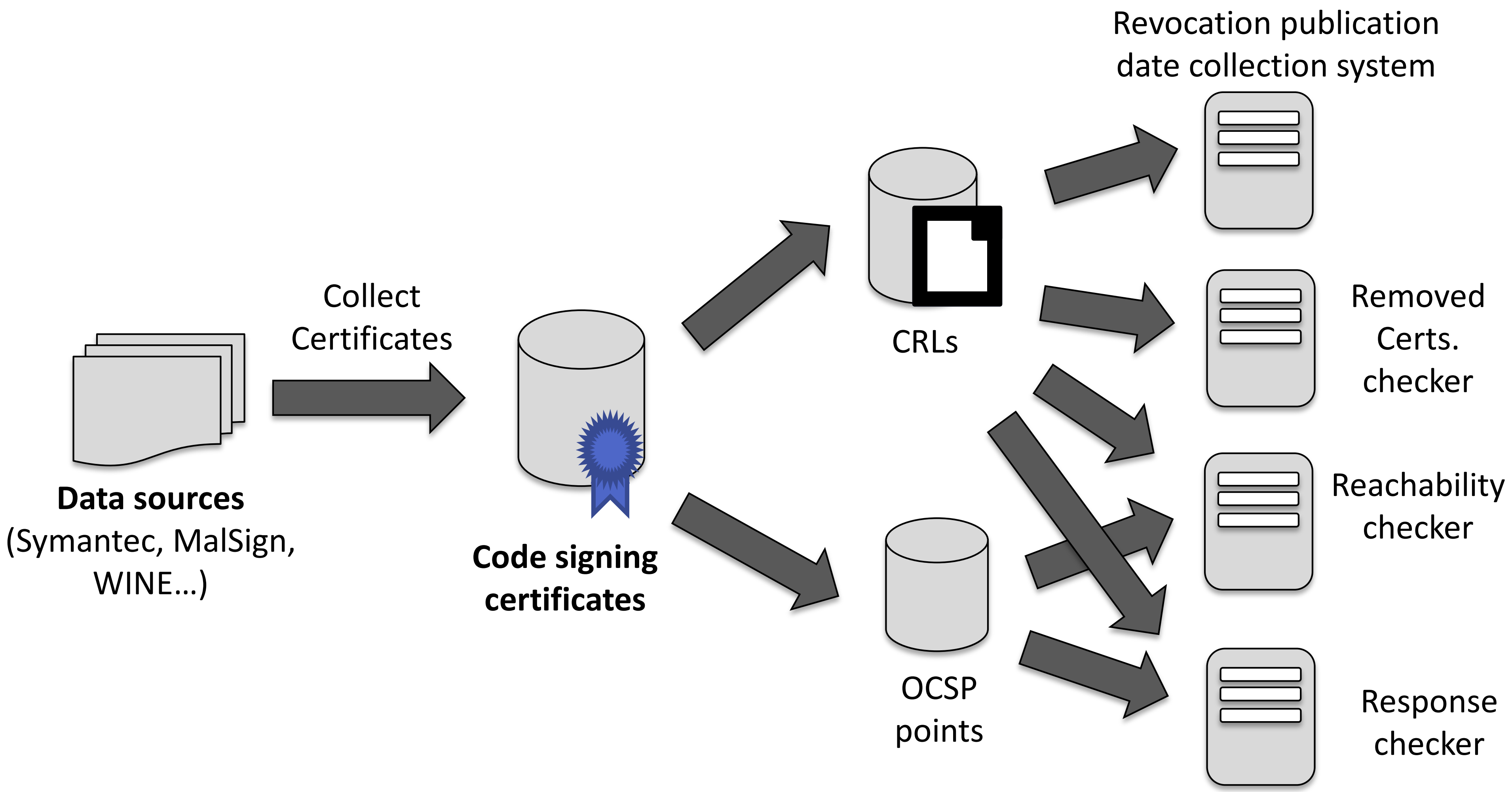
Data sources
(Symantec, MalSign, WINE…)

Code signing certificates

Data Collection



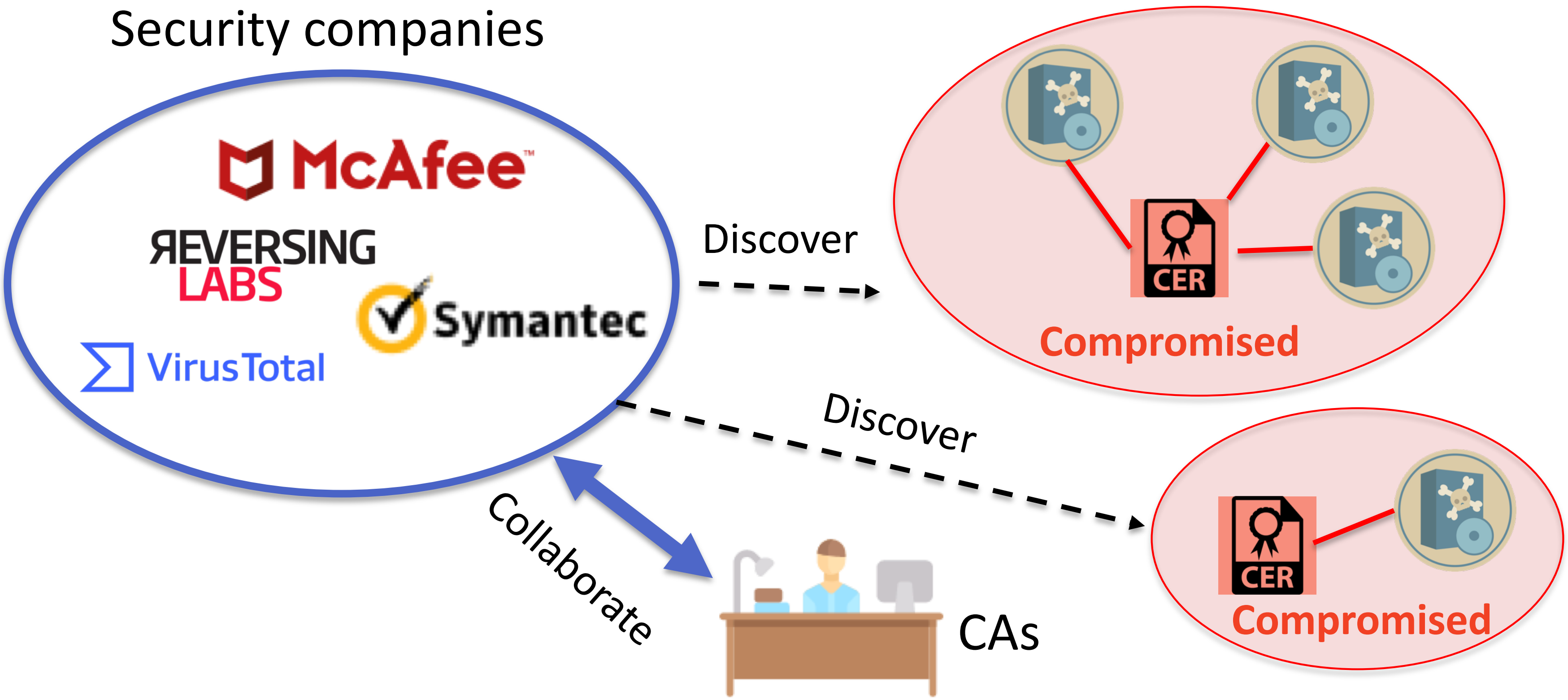
Data Collection



Outline

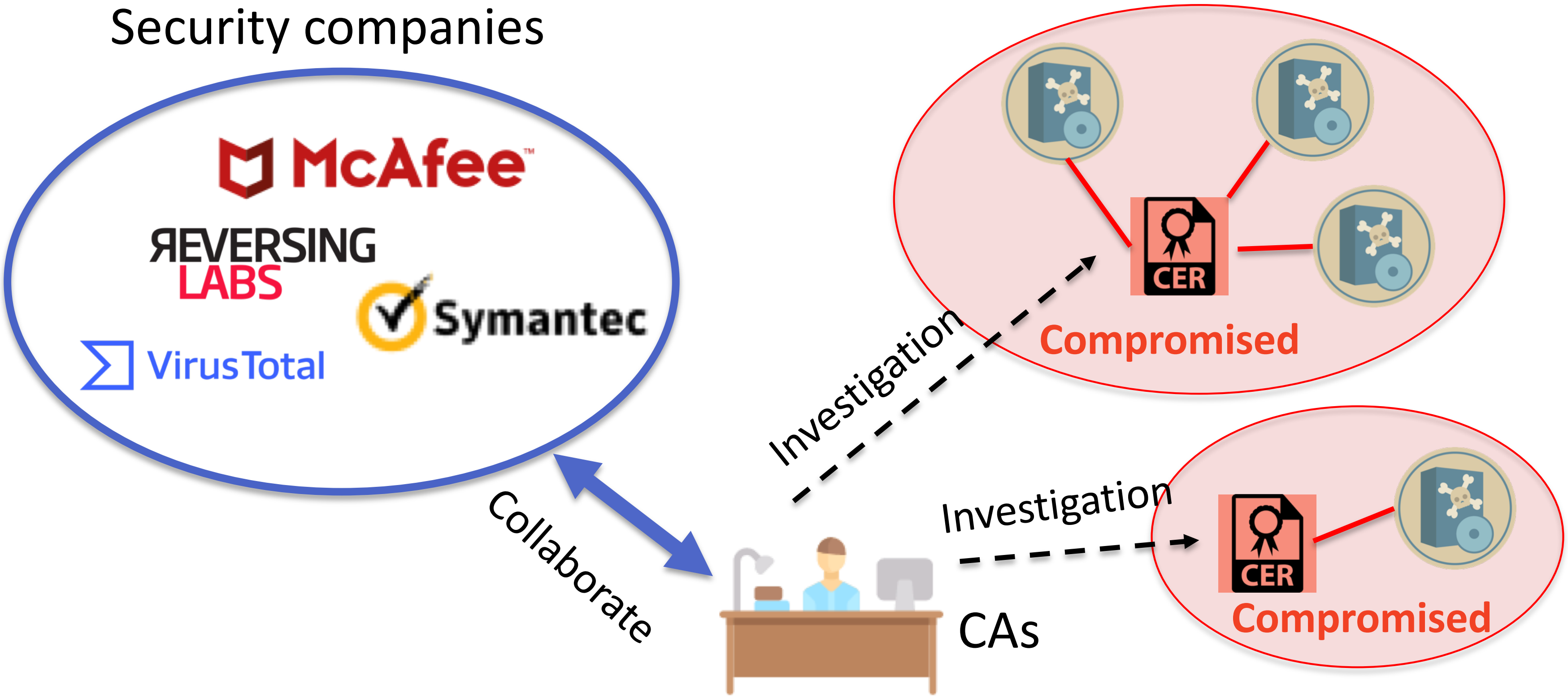
- Data collection
- Results: Effectiveness of revocation process
  - **Discovery of compromised certificates**
  - Invalidation of all signed malware
  - Dissemination of revocation information

Step #1: Discover Compromised Certificates



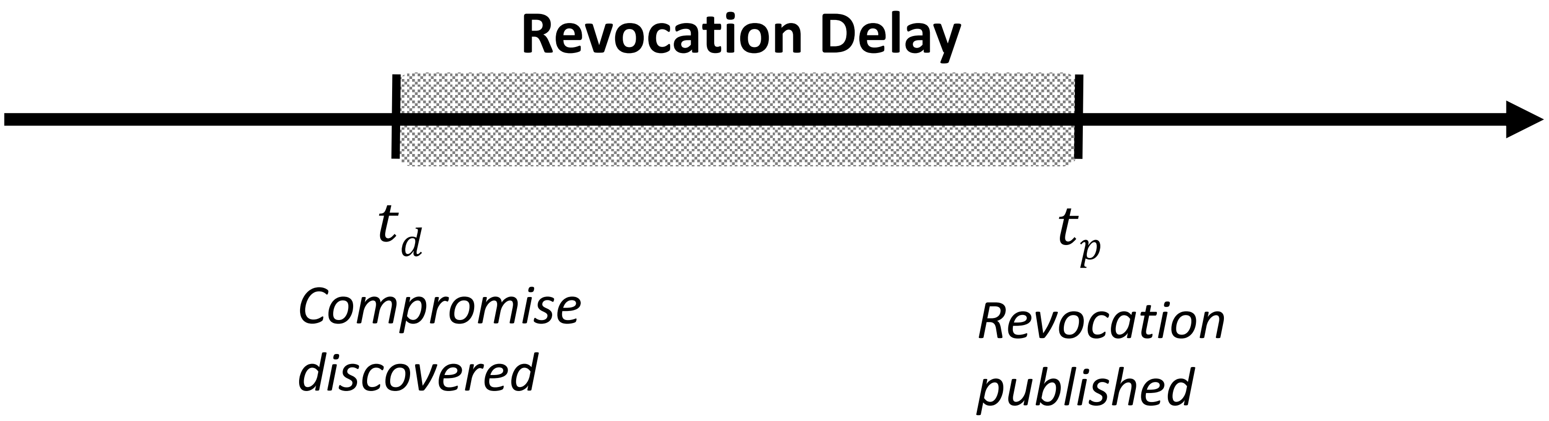
- Collaborate with security companies to promptly discover compromised certificates

Step #1: Discover Compromised Certificates



- Collaborate with security companies to promptly discover compromised certificates
- Promptly start investigations and revoke them
  - Revocation delay should be as short as possible

Revocation Delay: **Definition**



- Revocation delay: $t_p - t_d$
- t_d : the earliest detection dates of signed malware
  - E.g., the earliest submission date of VirusTotal
- t_p : the dates when revoked serial numbers are added to CRLs (aka *revocation publication date*)

Revocation Delay: **Result**

- Delay (t_p - t_d) : from 1 day to 1,553 days (4.25 years)

- Average delay: 171.4 days (5.6 months)

- Compromised certificates **not promptly revoked**

➔ Clients remain **exposed to this threat** for 5 months

Estimation of Compromised Certificates

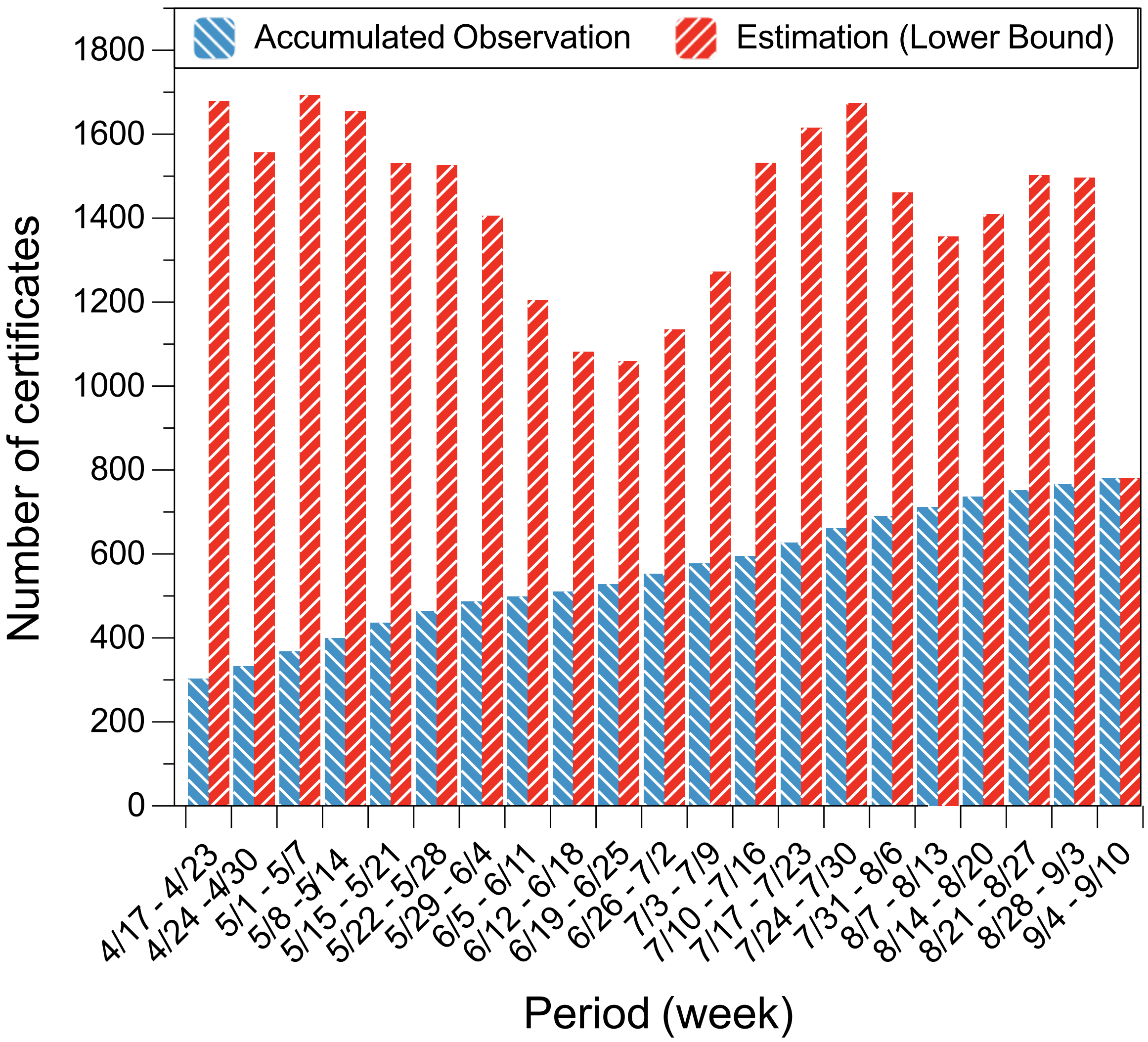
- Estimate the # of abused certificates in the wild
  - Used the mark-recapture methodology
  - Due to no corpus of code signing certificates to cover all code signing certificates in the wild

$$N = \frac{n1 * n2}{p}$$

P: Intersection of two samples
N1: sample #1
N2: sample #2

- Population:
  - n1: VirusTotal hunting data set
  - n2: Symantec telemetry data set

Discovery of Compromised Certificates



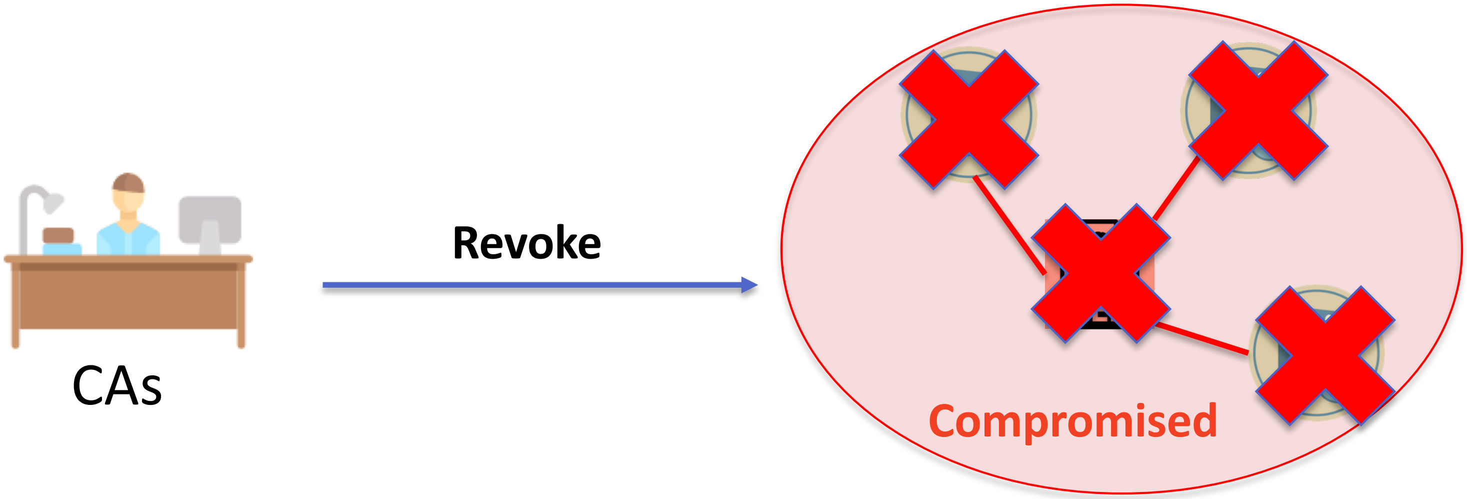
- Estimated compromised certificates are **2.74X larger** than actually observed
- Even large security companies **cannot cover most of compromised certificates** in the wild
  - A cause of **long revocation delay**

Outline

- Data collection
- Effectiveness of revocation process
  - Discovery of compromised certificates
  - **Invalidation of all signed malware**
  - Dissemination of revocation information

Role in the Second Step

- CAs should decide the ***effective revocation dates*** (t_r) to invalidate all malware signed with the compromised certificate



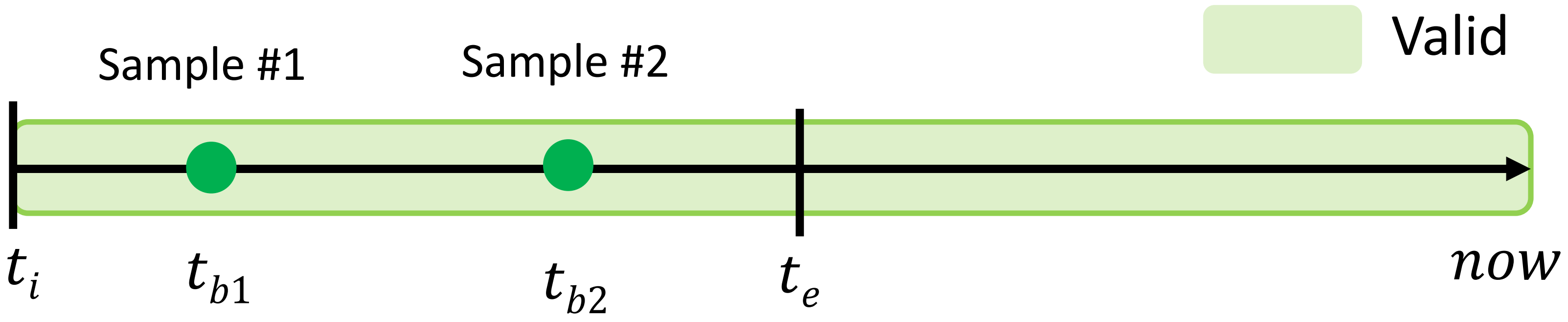
What is the Effective Revocation Dates (t_r)?

- Revocation will be made dependent on a specific date, **effective revocation date** (t_r)

- It determines the validity of signed samples
  - Depending on t_r signed samples become valid or invalid

What is the Effective Revocation Dates (t_r)?

- Revocation will be made dependent on an effective revocation date (t_r)

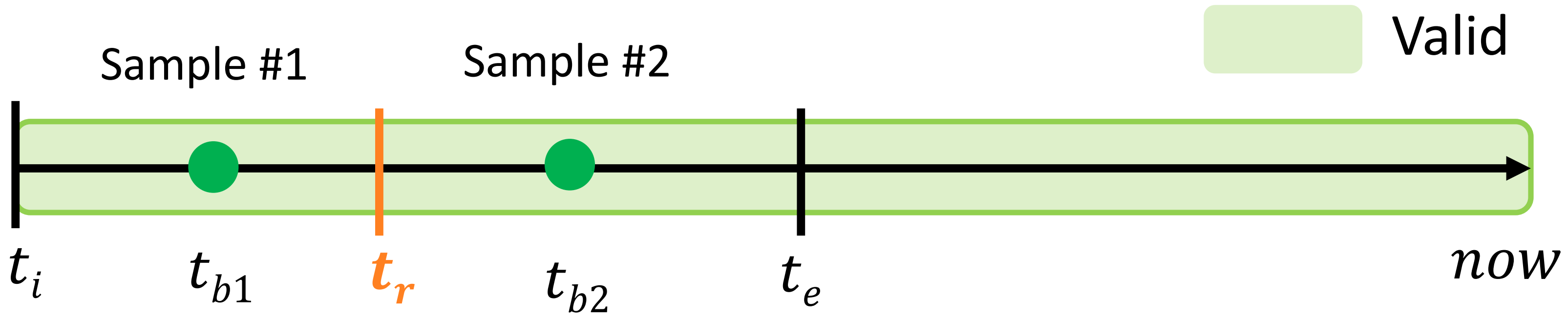


t_i: issue date

t_e: expiration date

What is the Effective Revocation Dates (t_r)?

- Revocation will be made dependent on an effective revocation date (t_r)

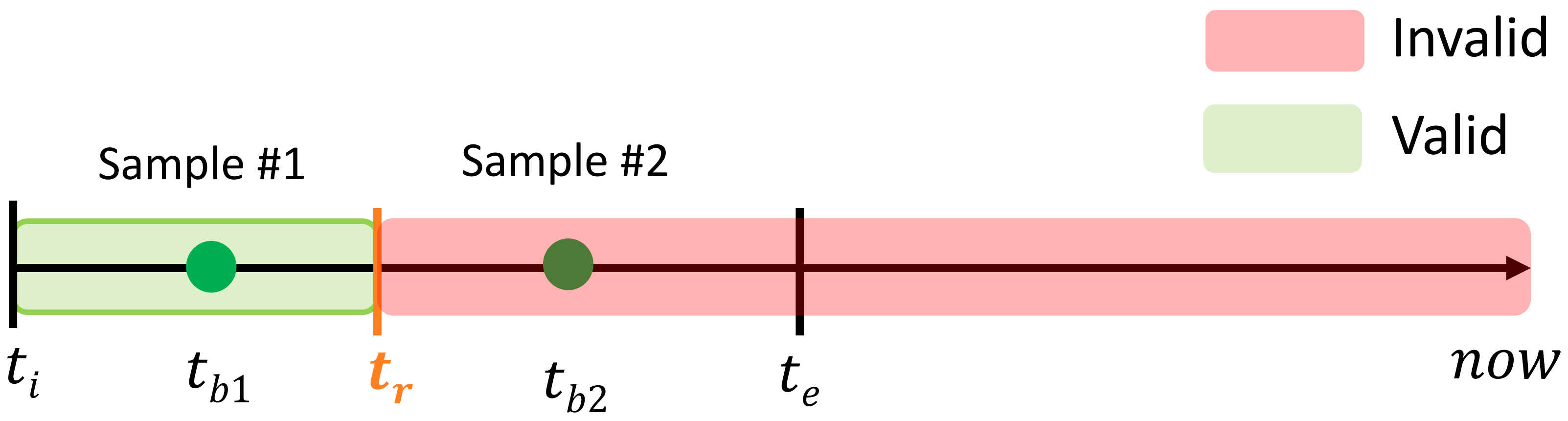


t_i: issue date

t_e: expiration date

What is the Effective Revocation Dates (t_r)?

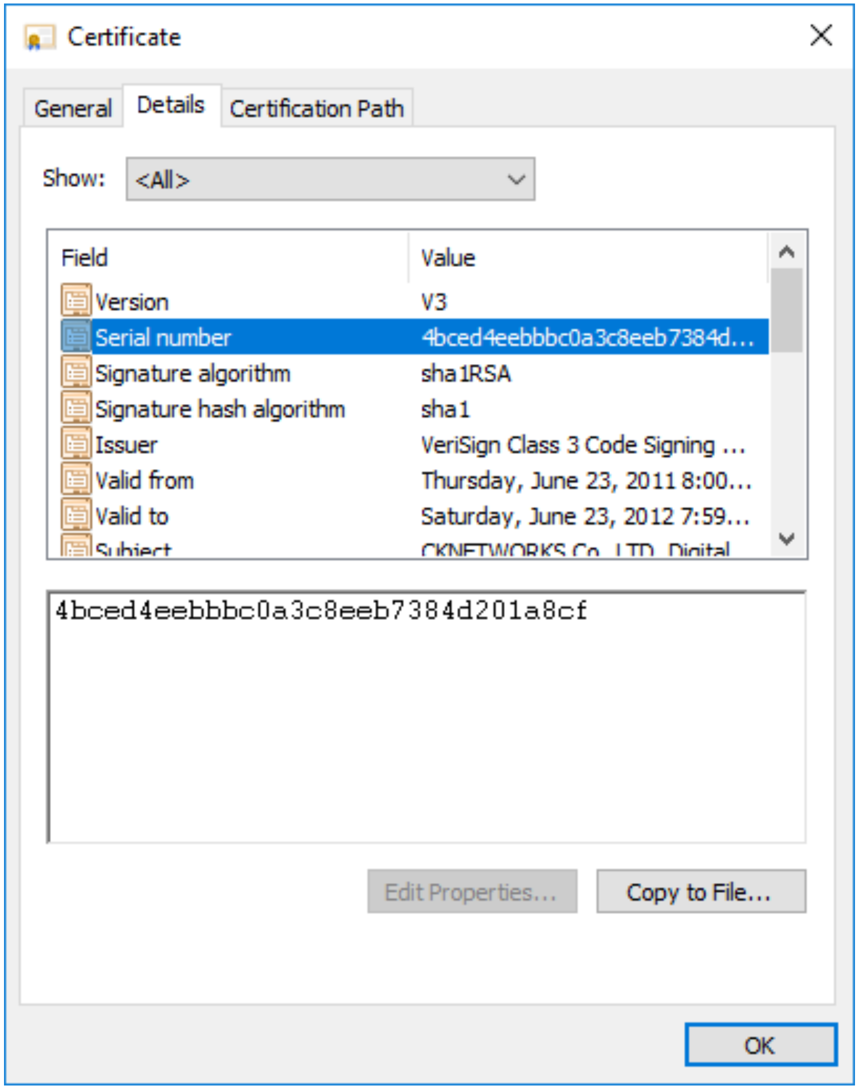
- Revocation will be made dependent on an effective revocation date (t_r)



t_i: issue date

t_e: expiration date

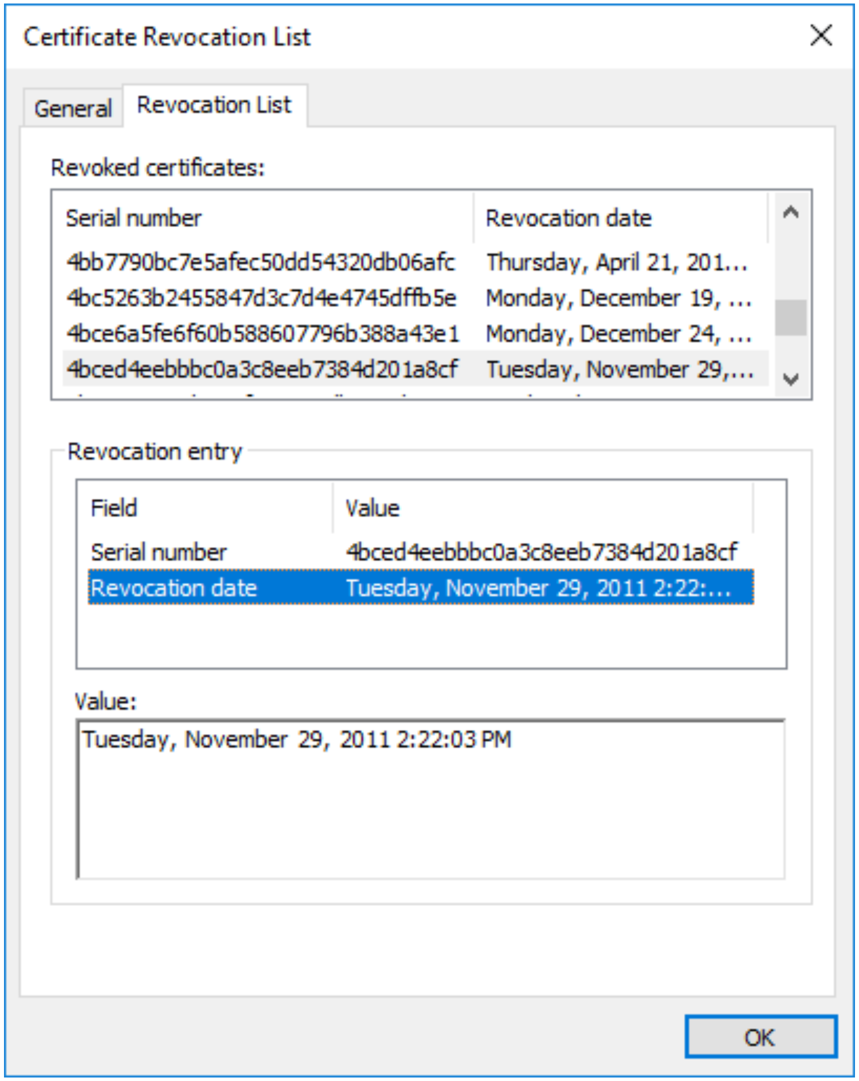
What is the Effective Revocation Dates (t_r)?



Show: <All>

Field	Value
Version	V3
Serial number	4bced4eebbbc0a3c8eeb7384d...
Signature algorithm	sha1RSA
Signature hash algorithm	sha1
Issuer	VeriSign Class 3 Code Signing ...
Valid from	Thursday, June 23, 2011 8:00...
Valid to	Saturday, June 23, 2012 7:59...
Subject	CKNETWORKS Co., LTD. Digital

```
4bced4eebbbc0a3c8eeb7384d201a8cf
```



Revoked certificates:

Serial number	Revocation date
4bb7790bc7e5afec50dd54320db06afc	Thursday, April 21, 201...
4bc5263b2455847d3c7d4e4745dffb5e	Monday, December 19, ...
4bce6a5fe6f60b588607796b388a43e1	Monday, December 24, ...
4bced4eebbbc0a3c8eeb7384d201a8cf	Tuesday, November 29,...

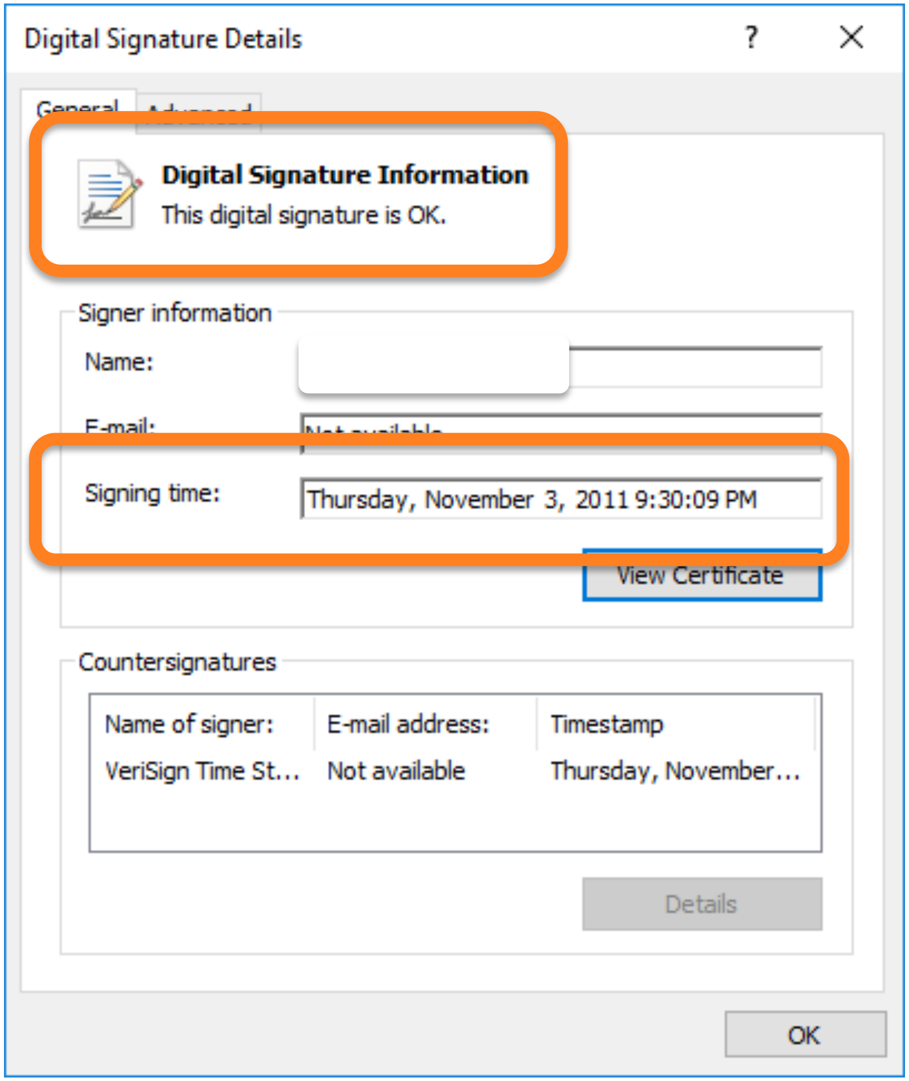
Revocation entry

Field	Value
Serial number	4bced4eebbbc0a3c8eeb7384d201a8cf
Revocation date	Tuesday, November 29, 2011 2:22:...

Value:

Tuesday, November 29, 2011 2:22:03 PM

What is the Effective Revocation Dates (t_r)?



Digital Signature Details

Digital Signature Information

This digital signature is OK.

Signer information

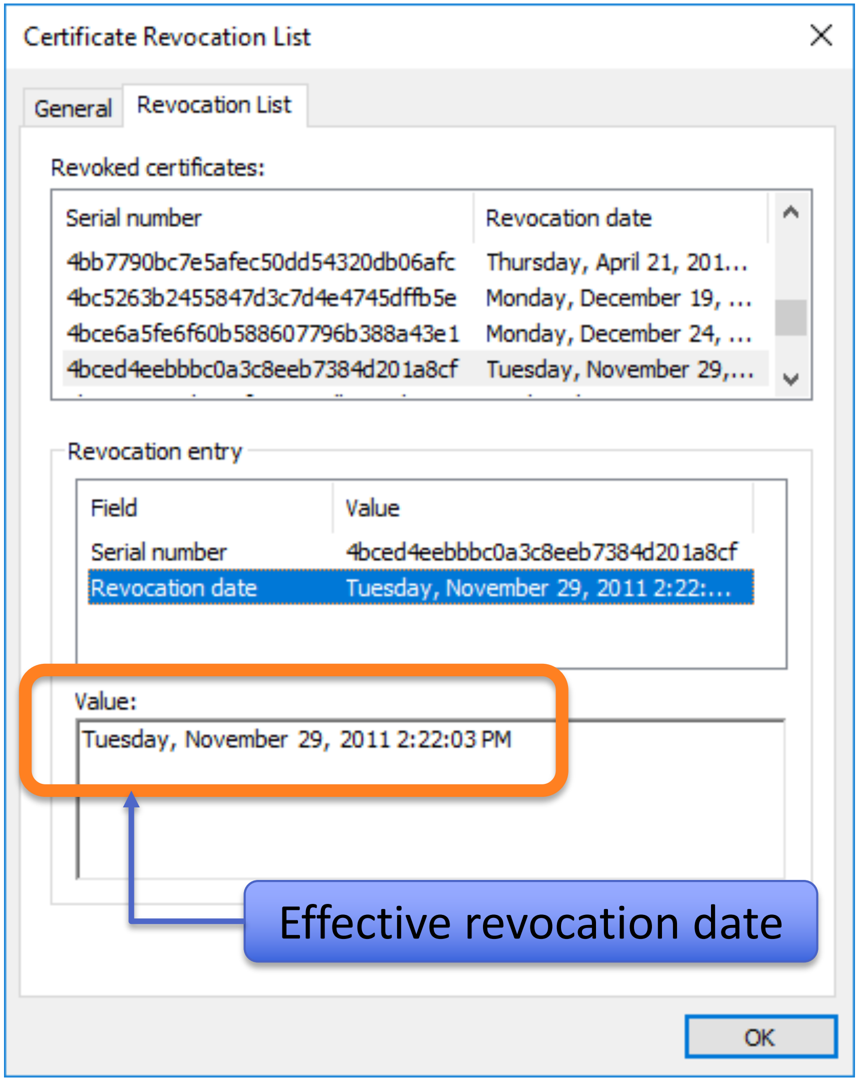
Name: [redacted]

E-mail: Not available

Signing time: Thursday, November 3, 2011 9:30:09 PM

Countersignatures

Name of signer:	E-mail address:	Timestamp
VeriSign Time St...	Not available	Thursday, November...



Certificate Revocation List

Revoked certificates:

Serial number	Revocation date
4bb7790bc7e5afec50dd54320db06afc	Thursday, April 21, 201...
4bc5263b2455847d3c7d4e4745dffb5e	Monday, December 19, ...
4bce6a5fe6f60b588607796b388a43e1	Monday, December 24, ...
4bced4eebbbc0a3c8eeb7384d201a8cf	Tuesday, November 29,...

Revocation entry

Field	Value
Serial number	4bced4eebbbc0a3c8eeb7384d201a8cf
Revocation date	Tuesday, November 29, 2011 2:22:...

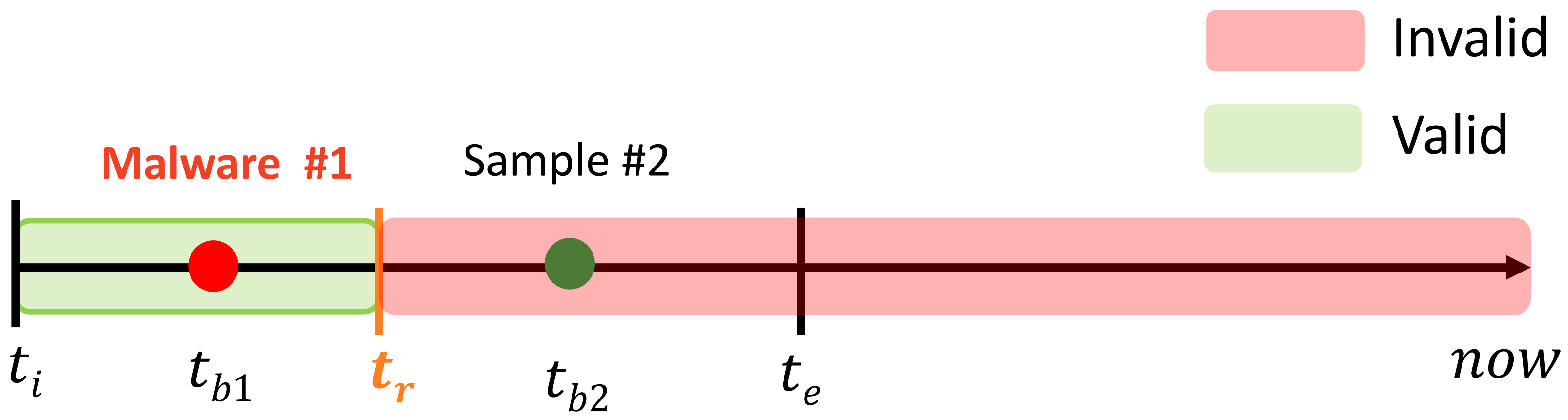
Value:

Tuesday, November 29, 2011 2:22:03 PM

Effective revocation date

Security Threat

- What if sample signed before t_r are malware?
  - Clients are exposed to the security threat



t_i: issue date

t_e: expiration date

Two Types of Revocation

- Soft revocation: $t_i < t_r < t_e$
  - Invalidate only samples signed after t_r
  - But security threats exist

- Hard revocation: $t_r = t_i$
  - No security threats, but invalidate all benign samples

Trends of Revocation Policy by CAs

	$< t_i$	$= t_i$	$\leq te$	$> t_e$	Total
Comodo	0	426	1,437	17	1,880
Thawte	0	74	1,055	39	1,168
Go Daddy	2	14	672	18	706
VeriSign	2	59	430	51	542
DigiCert	1	161	323	3	488
Starfield	0	3	153	2	158
Symantec	0	33	89	1	123
WoSign	0	57	17	0	74
StartCom	0	0	47	0	47
Certum	0	1	9	0	10
Other	0	96	117	1	214
Total	5	924	4,349	132	5,410

- The majority is soft revocation (83%)

Trends of Revocation Policy by CAs

	$< t_i$	$= t_i$	$\le t_e$	$> t_e$	Total
Comodo	0	426	1,437	17	1,880
Thawte	0	74	1,055	39	1,168
Go Daddy	2	14	672	18	706
VeriSign	2	59	430	51	542
DigiCert	1	161	323	3	488
Starfield	0	3	153	2	158
Symantec	0	33	89	1	123
WoSign	0	57	17	0	74
StartCom	0	0	47	0	47
Certum	0	1	9	0	10
Other	0	96	117	1	214
Total	5	924	4,349	132	5,410

- The majority is soft revocation (83%)
- 132 (2.5%) certificates are set to after expiration date
  - **Ineffective** revocation
  - All signed samples still valid

Ineffective Revocation Date Setting

- 1,022 certificates, revoked out of 45,613 certificates

- Soft revocation: 891 (87%) certificates

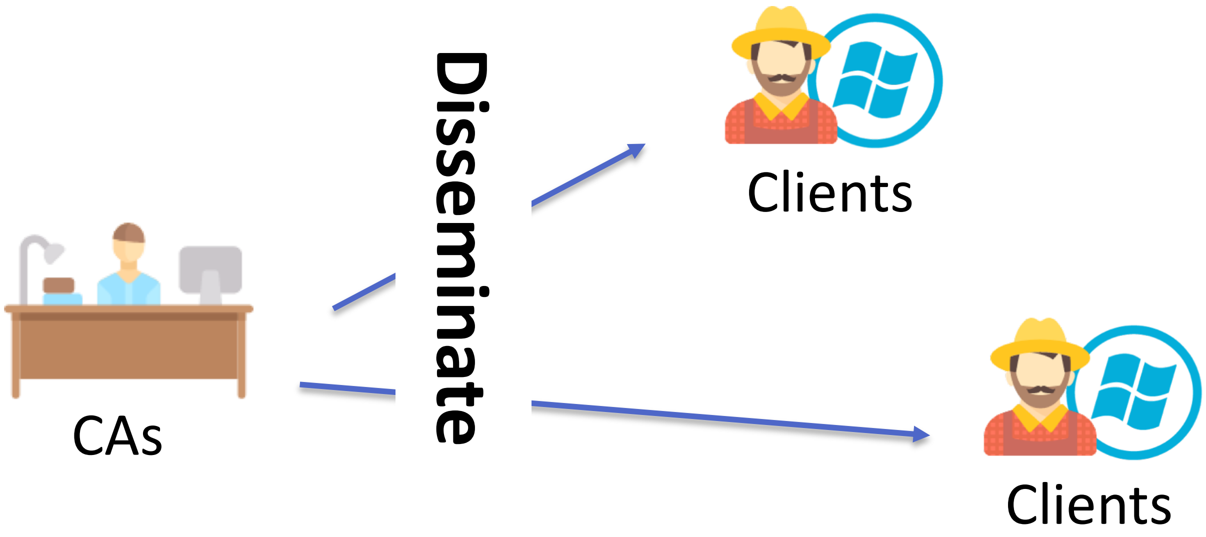
- Wrong effective revocation date: 45 (5%) certificates
  - 4,716 malware signed with the 45 certificates
  - 250 (5%) signed malware is still valid

➔ **Clients remain exposed to the security threat**

Outline

- Data collection
- Effectiveness of revocation process
  - Discovery of compromised certificates
  - Invalidation of all signed malware
  - **Dissemination of revocation information**

Roles in the Third Step



1. Specify CRLs and OCSP points in certificates
2. Responsible for expired certificates
3. Maintain infrastructure to be always-available for clients

Enforcement in Windows

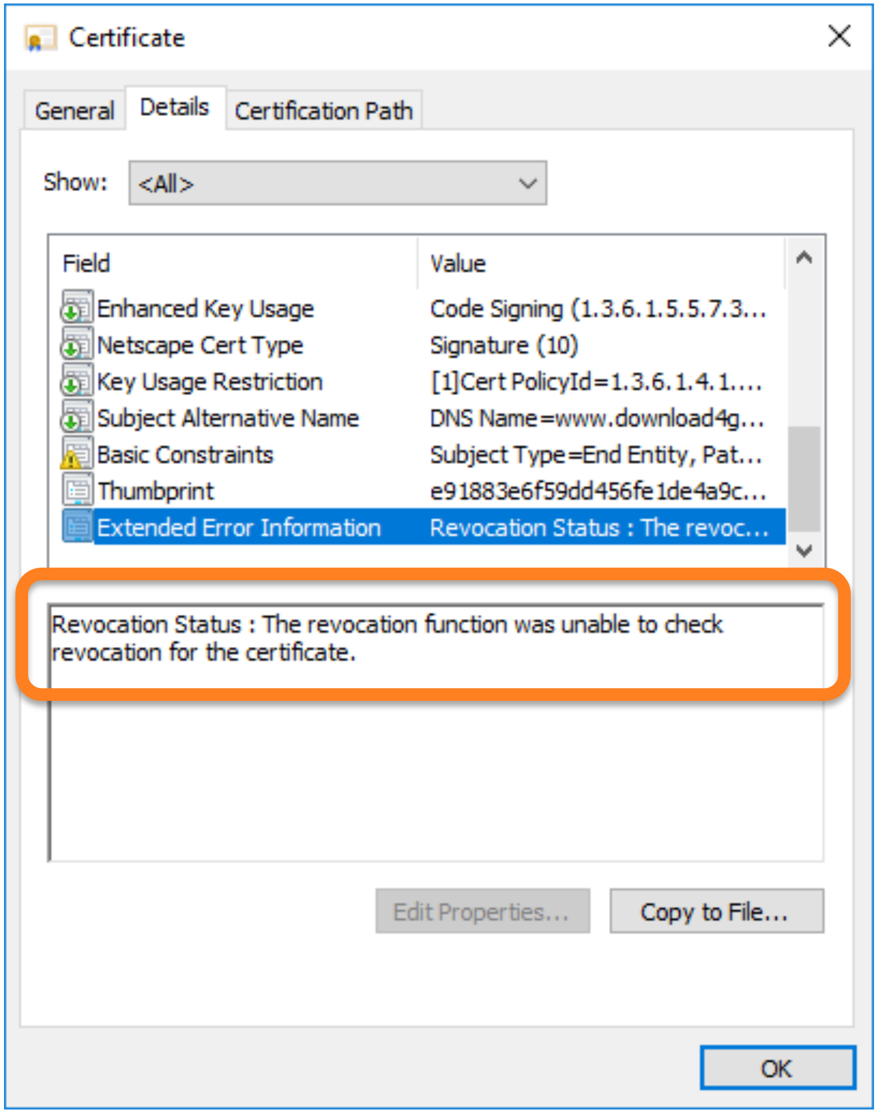
- *Soft-fail* policy for checking revocation status
  - Windows believes a certificate is valid unless revocation status information is available

#1. Certificates without CRLs and OCSP Points

- 788 certificates (0.5% out 144k): **no CRLs and OCSP**
  - 86% of them were issued by Thawte before 2003
  - All of them already expired
  - However, if malware is signed with the certificates and trust-timestamped, the malware can be still valid

➔ Clients have **no means** to check the status

#1. Certificates without CRLs and OCSP Points

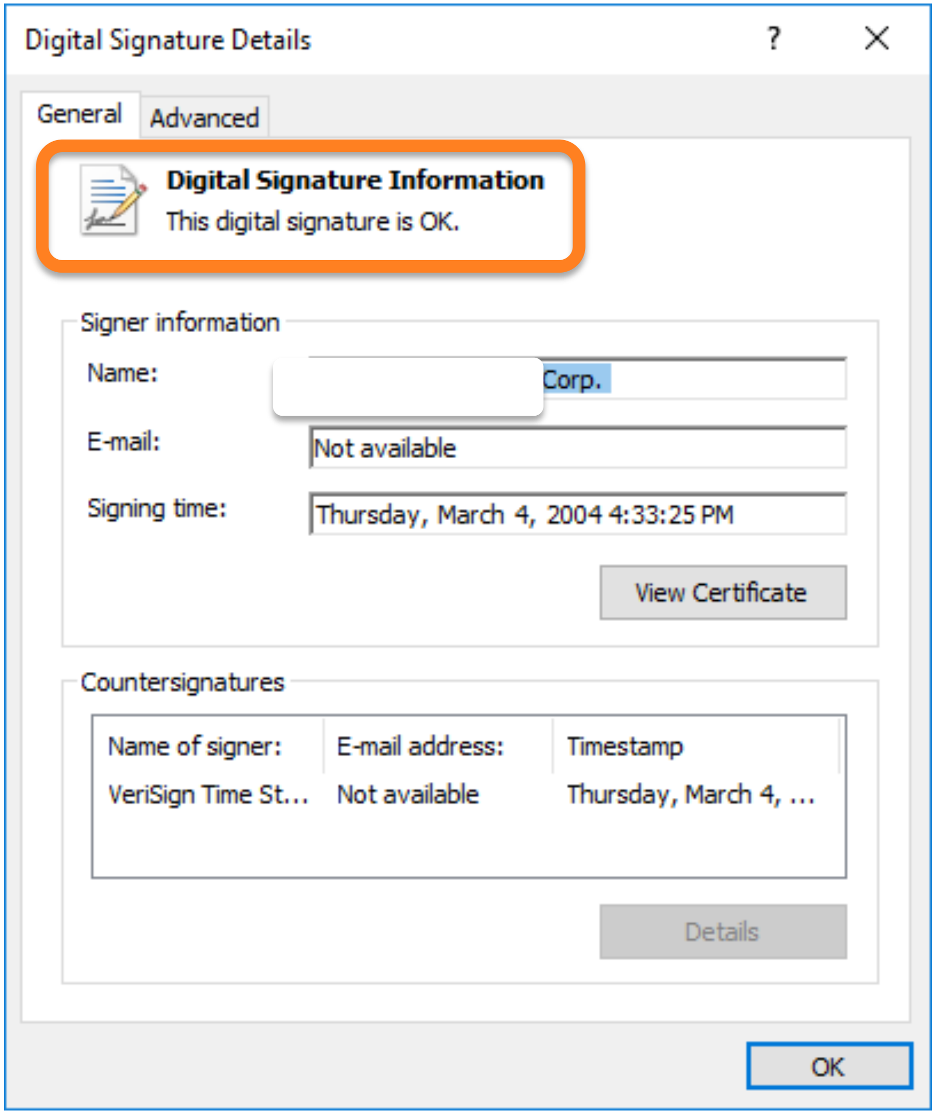


Certificate

Show: <All>

Field	Value
Enhanced Key Usage	Code Signing (1.3.6.1.5.5.7.3...
Netscape Cert Type	Signature (10)
Key Usage Restriction	[1]Cert PolicyId=1.3.6.1.4.1....
Subject Alternative Name	DNS Name=www.download4g...
Basic Constraints	Subject Type=End Entity, Pat...
Thumbprint	e91883e6f59dd456fe1de4a9c...
Extended Error Information	Revocation Status : The revoc...

Revocation Status : The revocation function was unable to check revocation for the certificate.



Digital Signature Details

Digital Signature Information

This digital signature is OK.

Signer information

Name: Corp.

E-mail: Not available

Signing time: Thursday, March 4, 2004 4:33:25 PM

Countersignatures

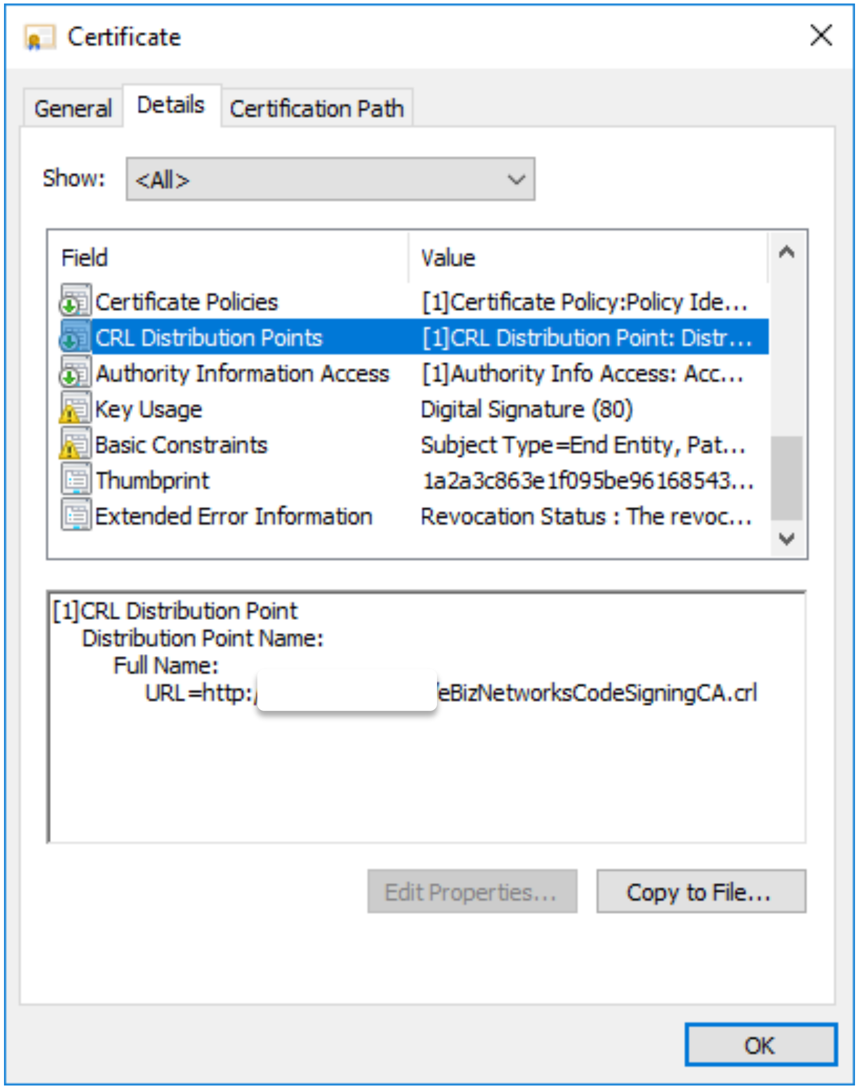
Name of signer:	E-mail address:	Timestamp
VeriSign Time St...	Not available	Thursday, March 4, ...

#2. Unreachable CRLs and OCSP Server

- 13 CRLs (6% out of 215) are unreachable
  - 5 CRLs: HTTP 404 Not Found error
    - They moved the CRLs file to another place
  - One CRL domain is taken by a domain reseller

- 15 OCSP points
  - Bad hostname, timeout, forbidden, & method not allowed

#2. Unreachable CRLs and OCSP Server

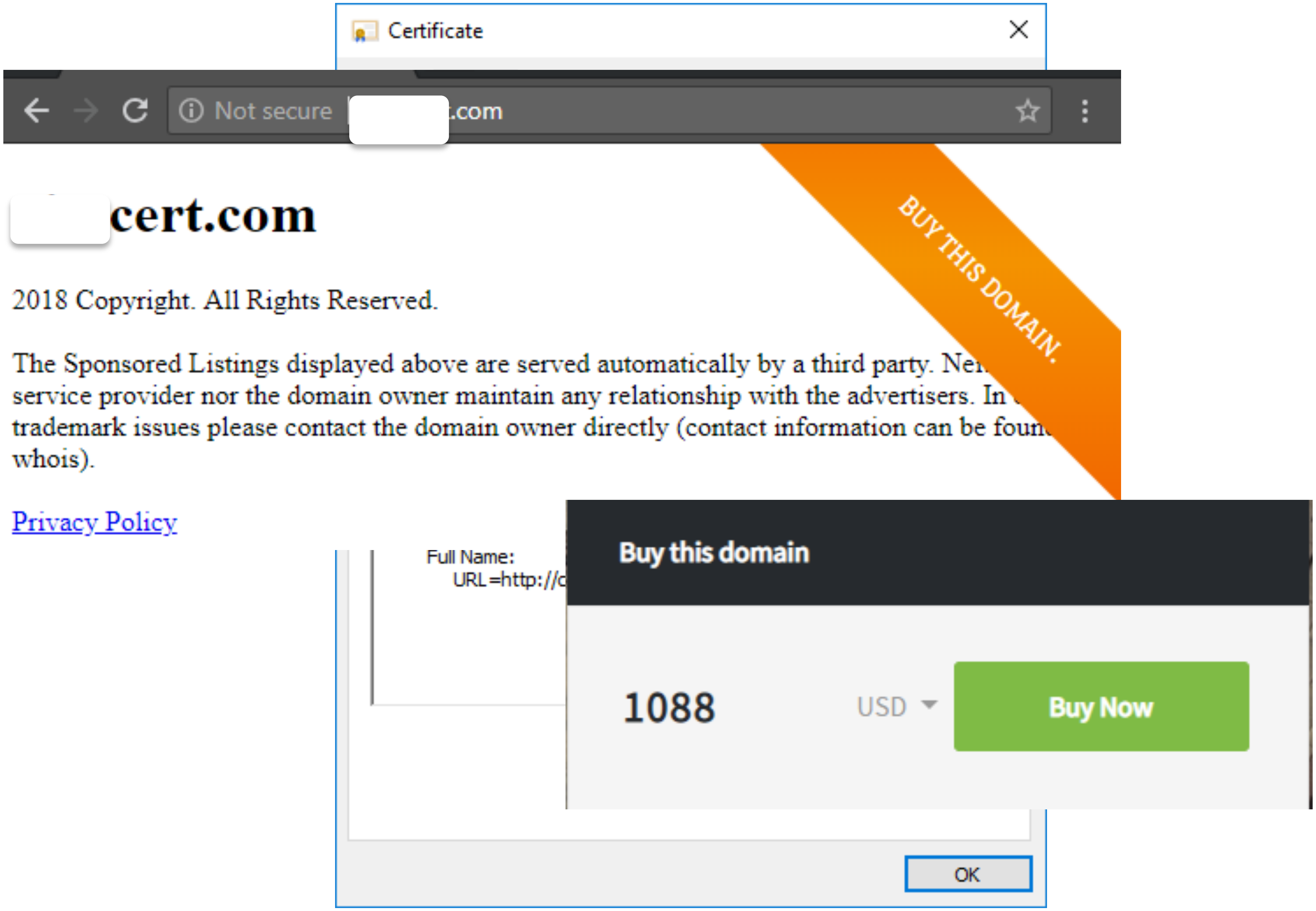


Show: <All>

Field	Value
Certificate Policies	[1]Certificate Policy:Policy Ide...
CRL Distribution Points	[1]CRL Distribution Point: Distr...
Authority Information Access	[1]Authority Info Access: Acc...
Key Usage	Digital Signature (80)
Basic Constraints	Subject Type=End Entity, Pat...
Thumbprint	1a2a3c863e1f095be96168543...
Extended Error Information	Revocation Status : The revoc...

```
[1]CRL Distribution Point
     Distribution Point Name:
          Full Name:
               URL=http:/[redacted]eBizNetworksCodeSigningCA.crl
```

#2. Unreachable CRLs and OCSP Server



Certificate

Not secure [redacted].com

[redacted]cert.com

2018 Copyright. All Rights Reserved.

The Sponsored Listings displayed above are served automatically by a third party. Ne
service provider nor the domain owner maintain any relationship with the advertisers. In
trademark issues please contact the domain owner directly (contact information can be foun
whois).

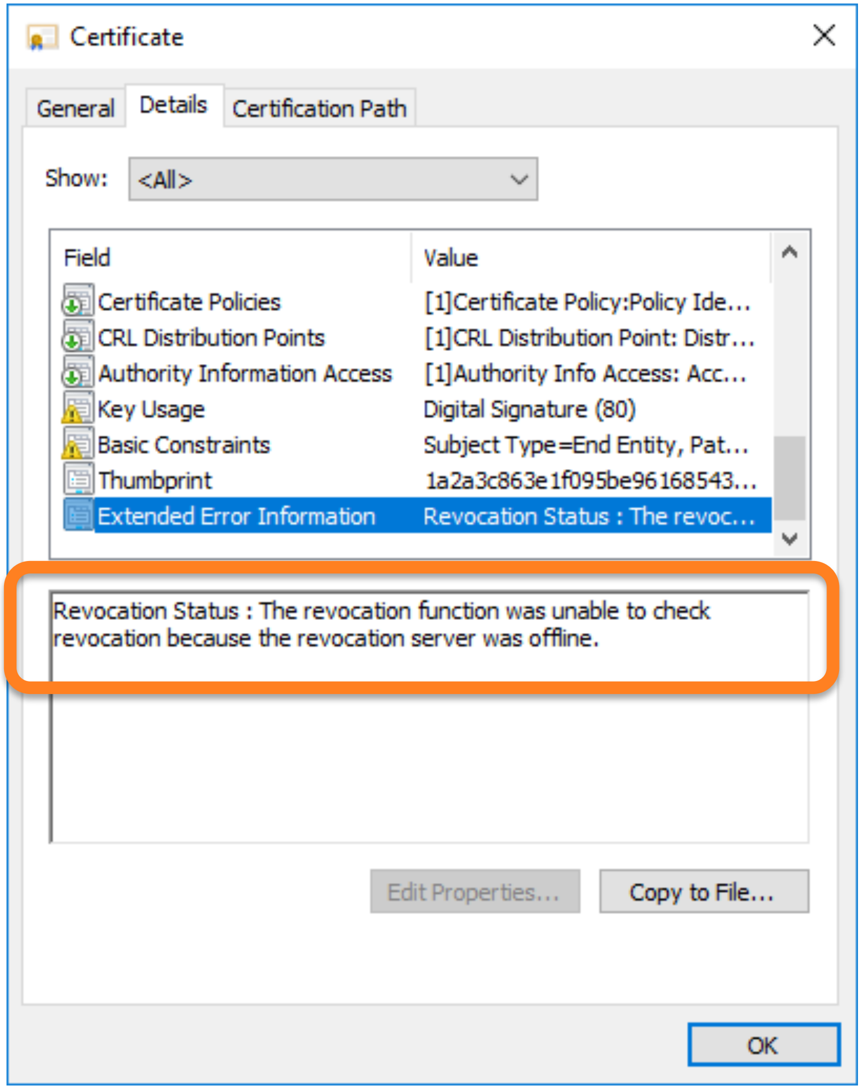
Privacy Policy

Full Name:
URL=http://c

Buy this domain

1088 USD

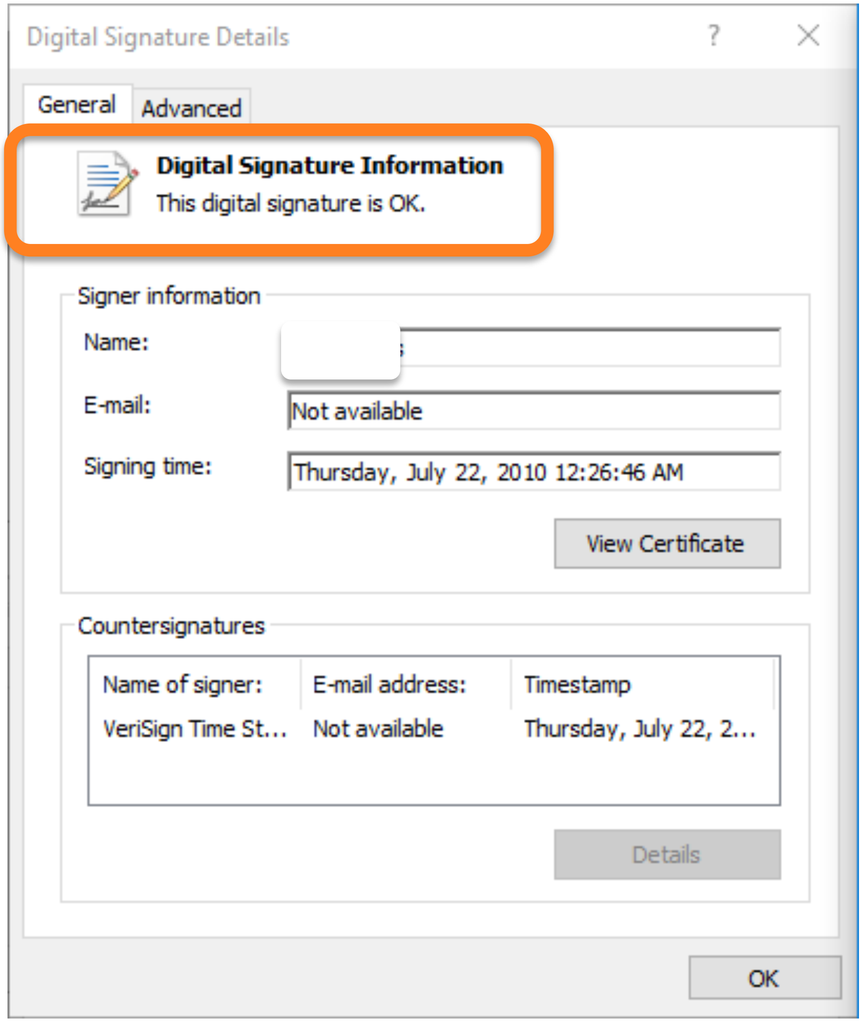
#2. Unreachable CRLs and OCSP Server



Show: <All>

Field	Value
Certificate Policies	[1]Certificate Policy:Policy Ide...
CRL Distribution Points	[1]CRL Distribution Point: Distr...
Authority Information Access	[1]Authority Info Access: Acc...
Key Usage	Digital Signature (80)
Basic Constraints	Subject Type=End Entity, Pat...
Thumbprint	1a2a3c863e1f095be96168543...
Extended Error Information	Revocation Status : The revoc...

Revocation Status : The revocation function was unable to check revocation because the revocation server was offline.



Digital Signature Information

This digital signature is OK.

Signer information

Name: [redacted]

E-mail: Not available

Signing time: Thursday, July 22, 2010 12:26:46 AM

Countersignatures

Name of signer:	E-mail address:	Timestamp
VeriSign Time St...	Not available	Thursday, July 22, 2...

#2. Unreachable CRLs and OCSP Server

- 13 CRLs (6% out of 215) are unreachable
  - 5 CRLs: HTTP 404 Not Found error
    - They moved the CRLs file to another place
  - One CRL domain is taken by a domain reseller

- 15 OCSP points
  - Bad hostname, timeout, forbidden, & method not allowed

➔Programs signed with the certificates can still **be valid**

- due to trust timestamping and *soft-fail* policy

#3-1. Transient Revoked Certificates in CRLs

- Recall: CAs, responsible for even expired certificates

- But, 278 revoked certificates **removed** from 18 CRLs

- Contacted the all CAs
  - A CA started investigations and found the flaw
  - And fixed the flaw thanks to our study and replied …
    - "Thank you … we were **removing** certificates from the CRL that had **expired** … We've modified our system to now exclude Code Signing, which means that **once revoked**, the certificate should **remain** on the CRL **indefinitely**."

➔ Even CAs **misunderstand** the code signing PKI

#3-2. Inconsistent Responses from CRLs and OCSP

- Responses from CRLs and OCSP should be consistent
  - E.g., if one is found in CRLs, the response from OCSP for the certificate indicates that “revoked”

- 19 certificates have **inconsistent** responses
  - All certificates were issued by Go Daddy and StartField

➔ CAs **improperly** maintain OCSP and CRLs servers

Conclusion

- The primary defense against abuse is **revocation**

- Revocation in code signing PKI is **more critical** and **more challenging** than TLS

- Hard to discover compromised certificates & samples

- Erroneously setting effective revocation dates
  - Makes malware valid although the certificate is revoked

- Improper dissemination of revocation information
  - Makes signed malware valid due to the *soft-fail* policy

Data Release

- Our data sets are available at **signedmalware.org**
  - CRLs for code signing certificates
  - Revocation publication dates

Thank you!

Doowon Kim

doowon@cs.umd.edu

http://signedmalware.org