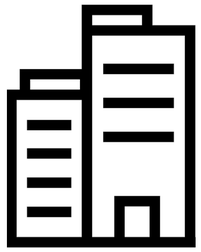
Tracing Your Roots: Exploring the TLS Trust Anchor Ecosystem

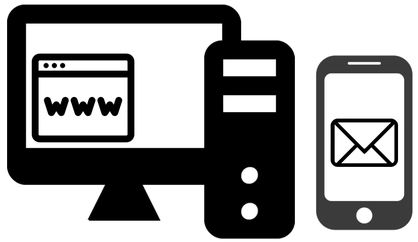
Zane Ma

Postdoc Researcher
Georgia Institute of Technology
zanema@gatech.edu
https://zanema.com

TLS + Web PKI



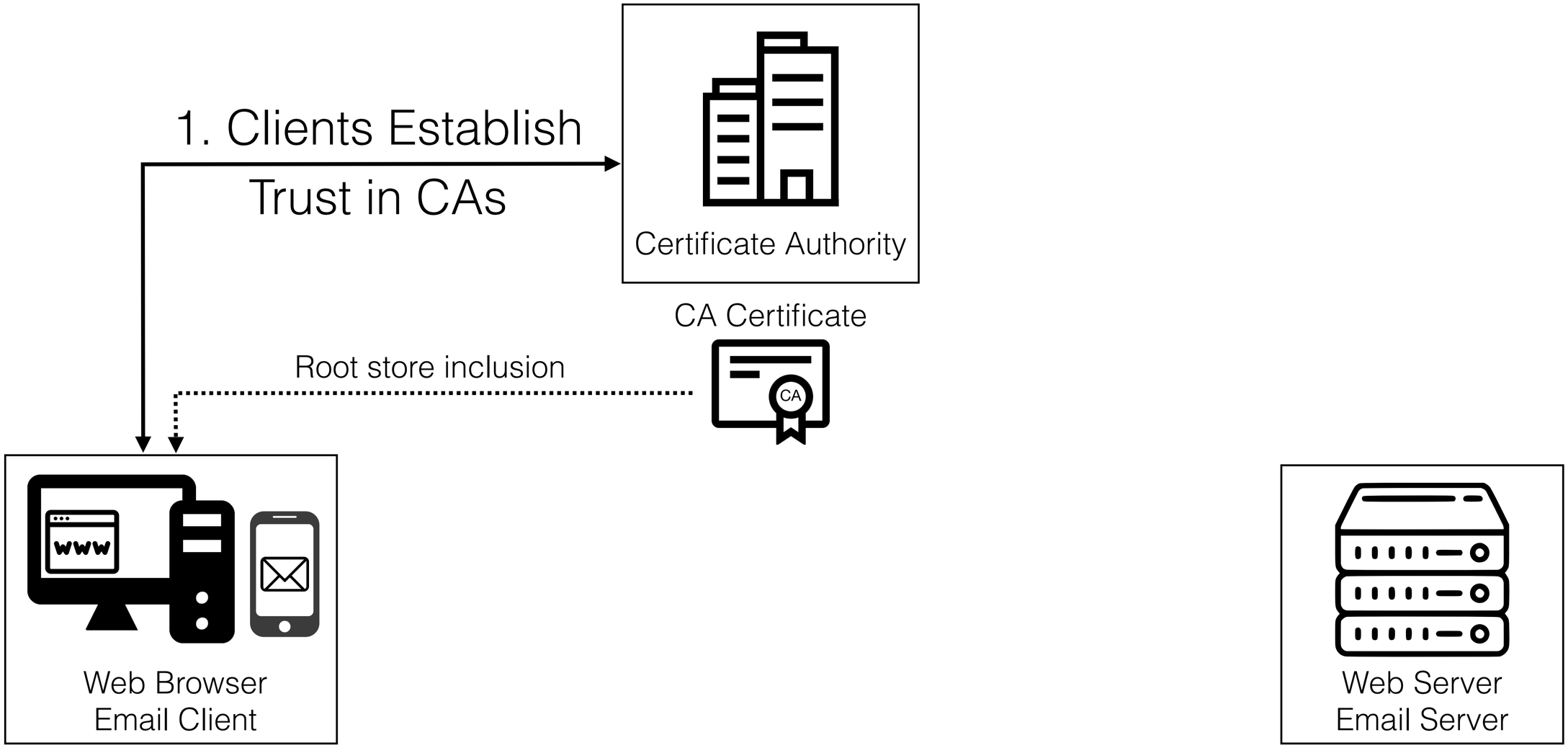
Certificate Authority



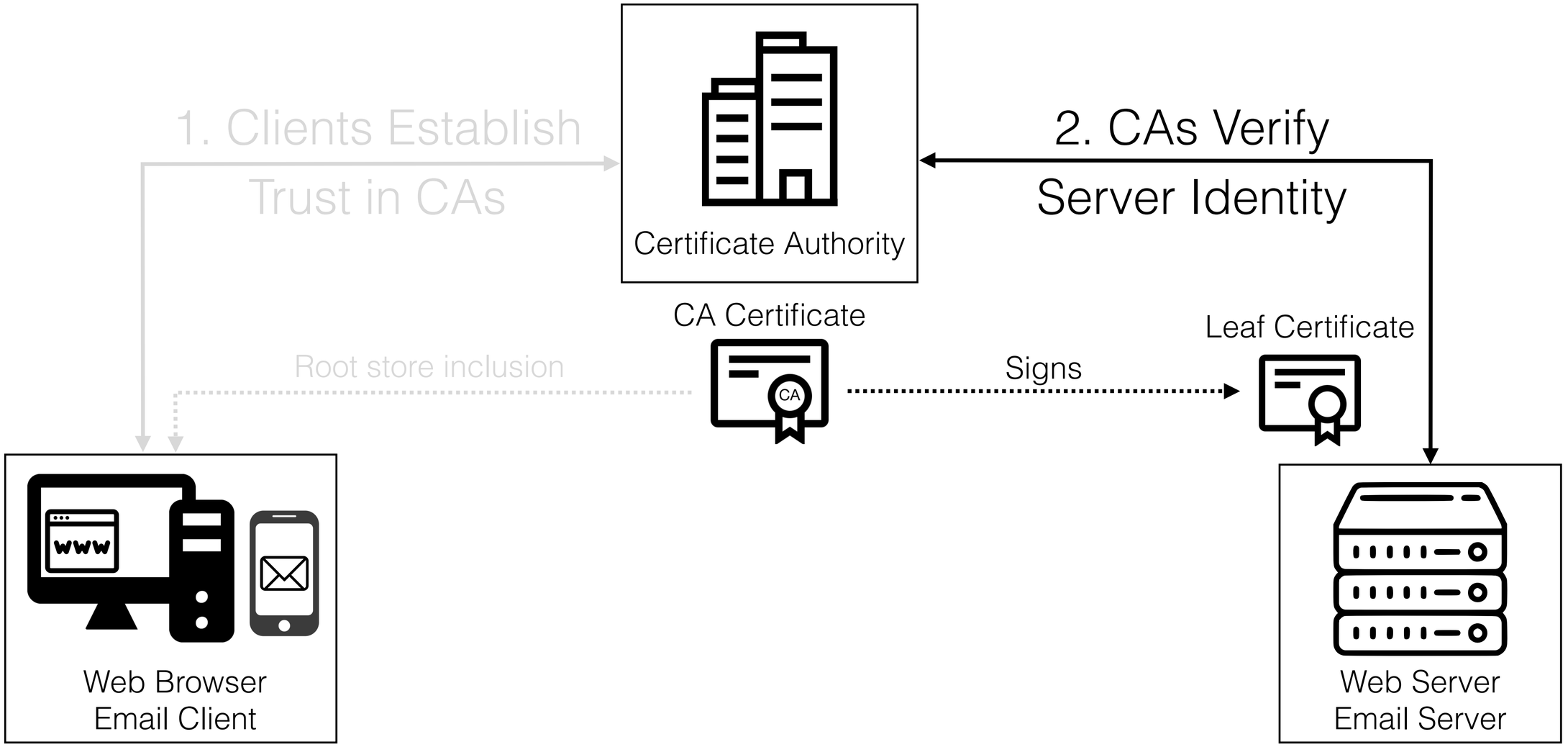
Web Browser
Email Client

Web Server
Email Server

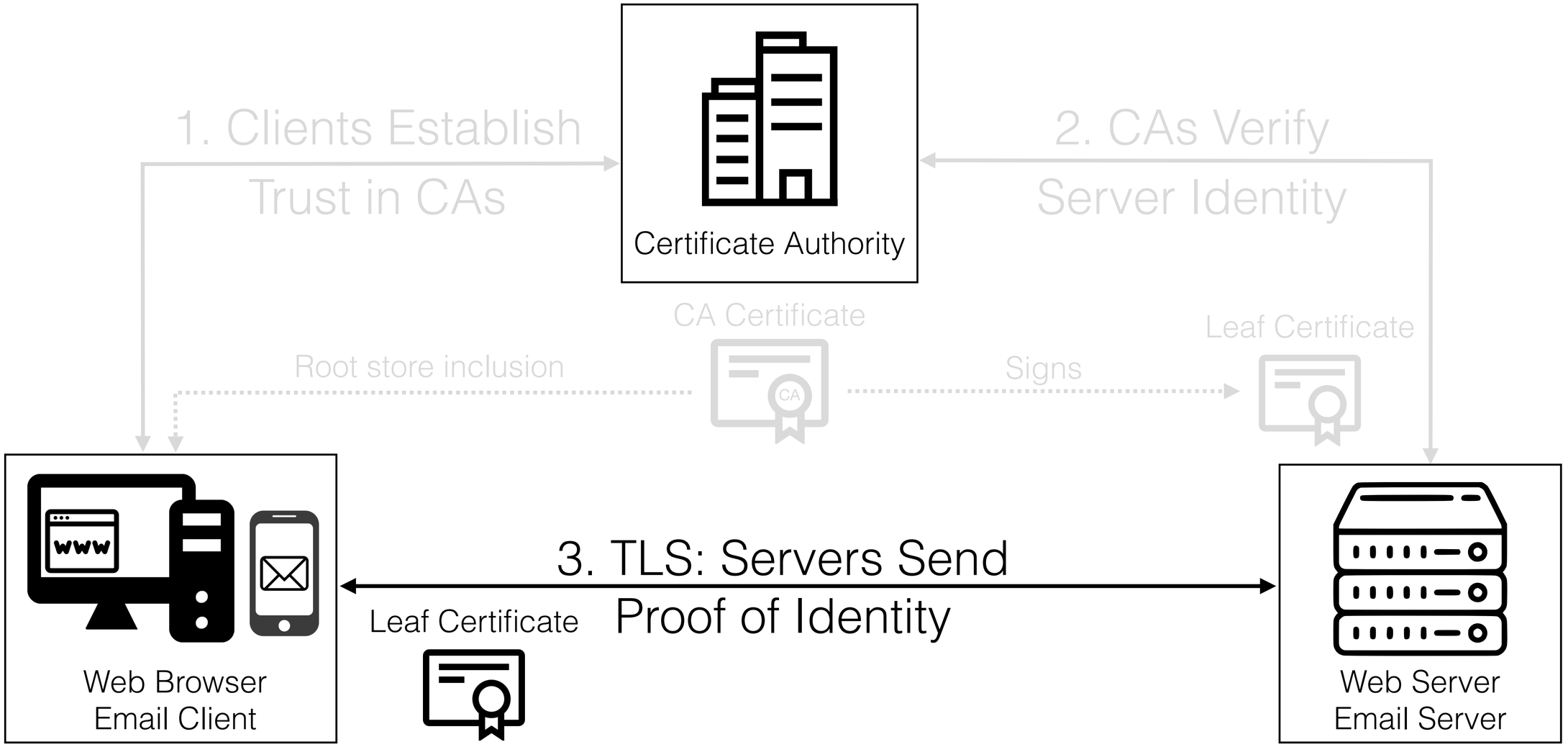
TLS + Web PKI


1. Clients Establish
Trust in CAs
Certificate Authority
CA Certificate
Root store inclusion
CA
www
Web Browser
Email Client
Web Server
Email Server

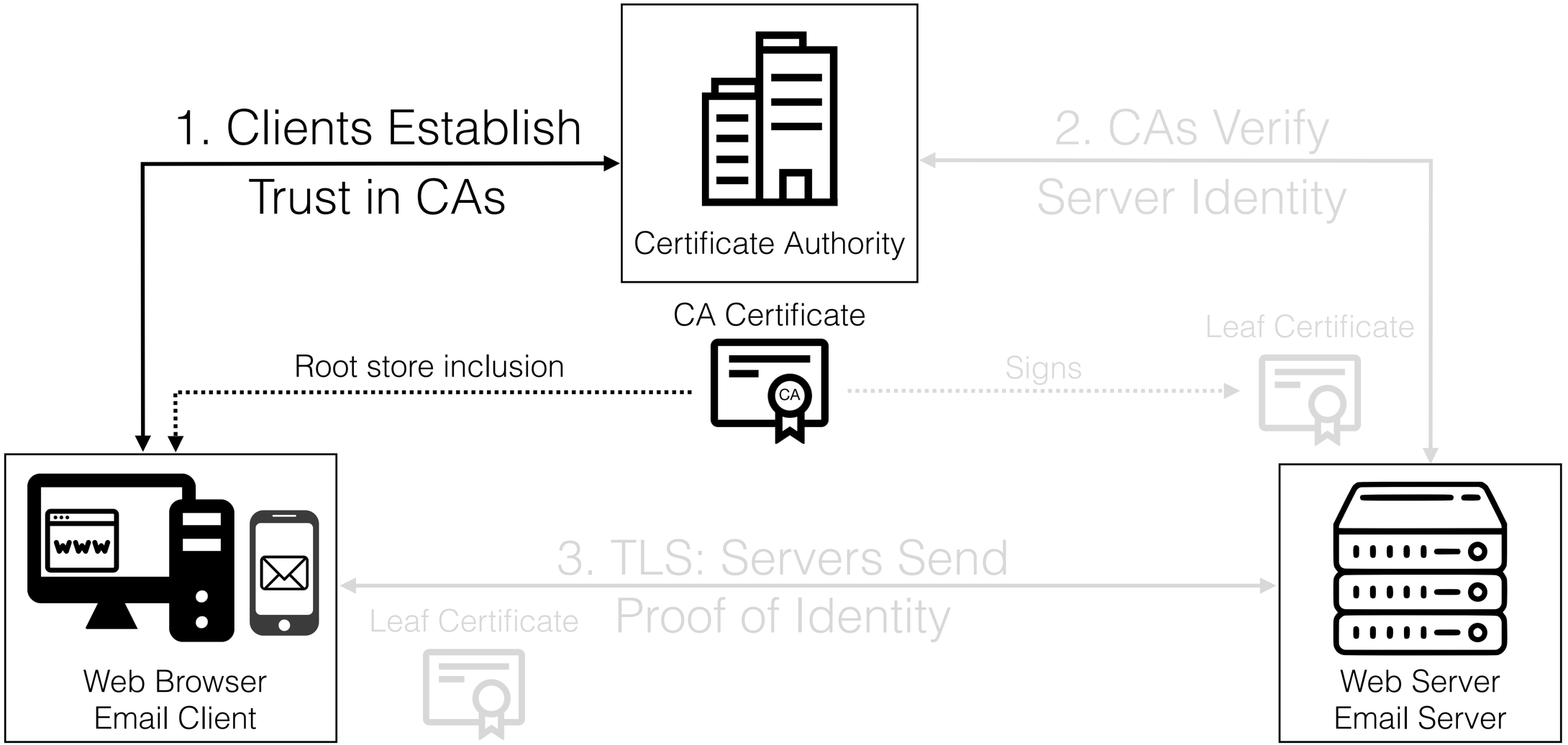
TLS + Web PKI


1. Clients Establish
Trust in CAs
Certificate Authority
2. CAs Verify
Server Identity
CA Certificate
Leaf Certificate
Root store inclusion
CA
Signs
www
Web Browser
Email Client
Web Server
Email Server

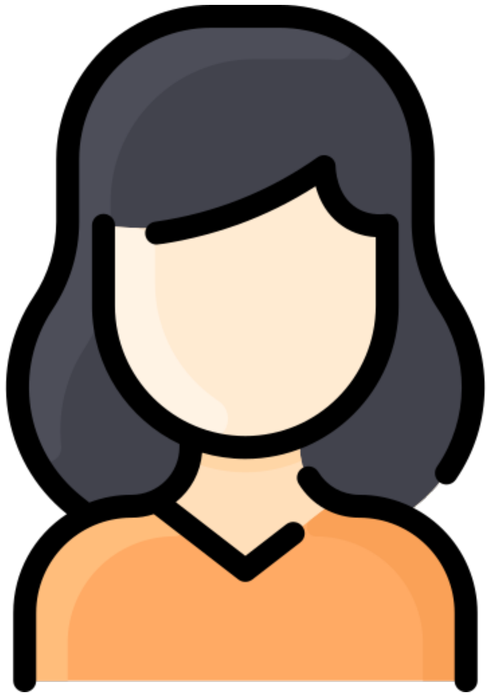
TLS + Web PKI

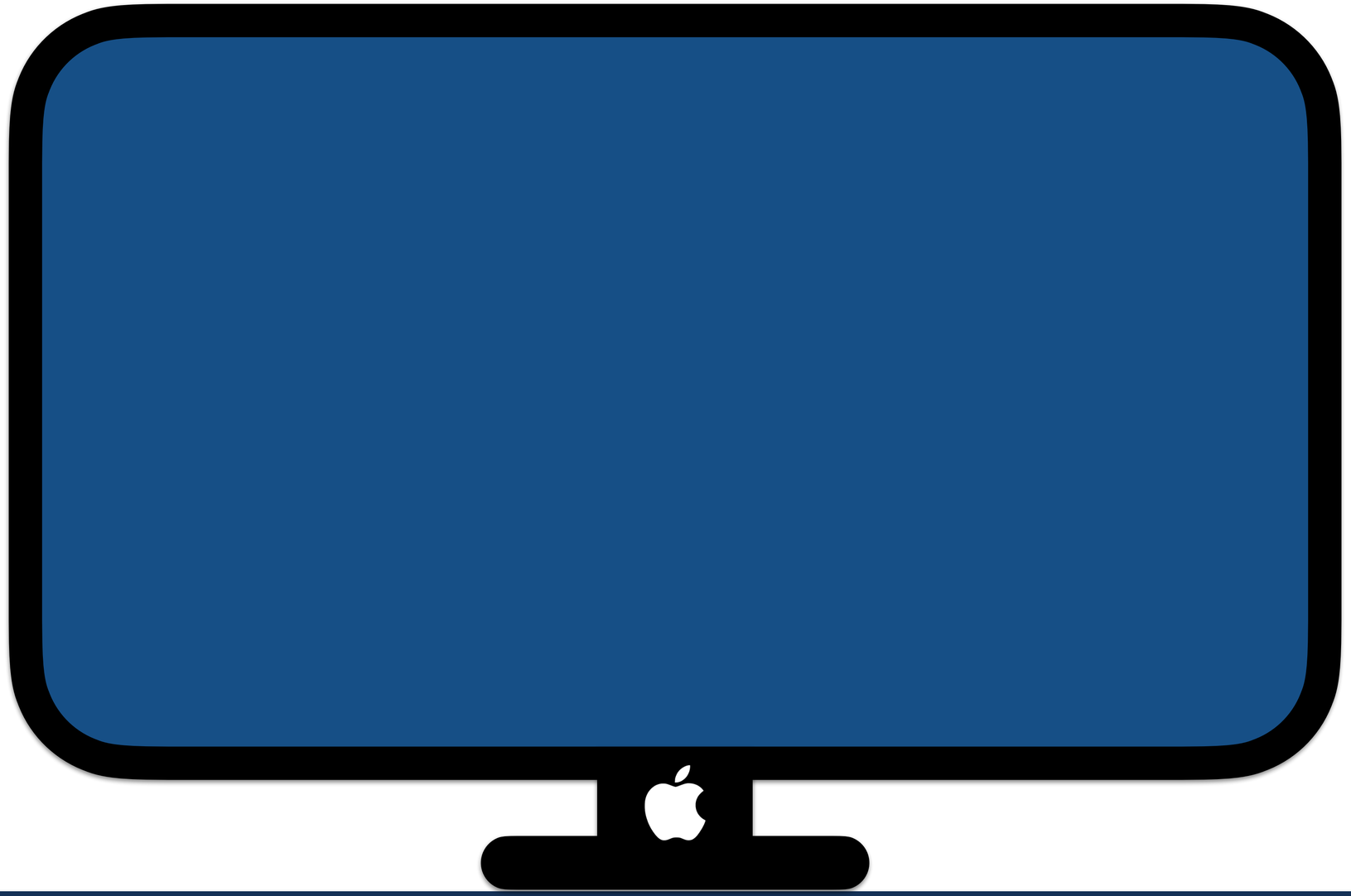

1. Clients Establish
Trust in CAs
2. CAs Verify
Server Identity
Certificate Authority
CA Certificate
Root store inclusion
CA
Signs
Leaf Certificate
www
Web Browser
Email Client
3. TLS: Servers Send
Proof of Identity
Leaf Certificate
Web Server
Email Server

TLS + Web PKI


1. Clients Establish
Trust in CAs
Certificate Authority
2. CAs Verify
Server Identity
CA Certificate
Root store inclusion
CA
Signs
Leaf Certificate
www
Web Browser
Email Client
3. TLS: Servers Send
Proof of Identity
Leaf Certificate
Web Server
Email Server

TLS user agents





TLS user agents

TLS user agents

TLS user agents

TLS user agents

TLS user agents

Research Questions

1. Which root store providers do TLS user agents rely on?

2. How do root store providers determine which CAs to trust?

3. Characterization of root store programs

4. How faithfully do providers copy root program trust?

Research Questions

1. Which root store providers do TLS user agents rely on?

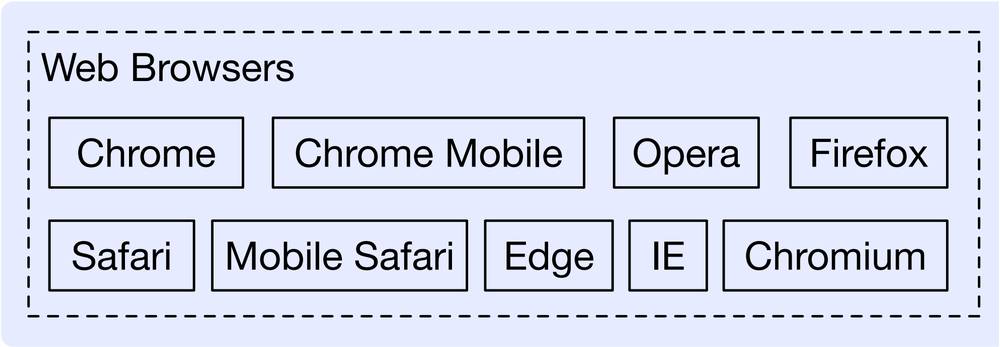
2. How do root store providers determine which CAs to trust?

3. Characterization of root store programs

4. How faithfully do providers copy root program trust?

Data collection

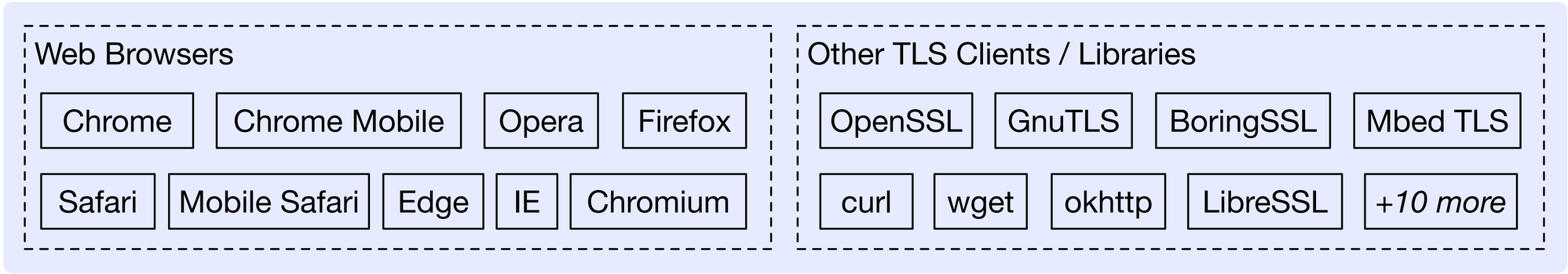
User Agents



Collected root stores for 77% of global CDN top 200 user agents

Data collection

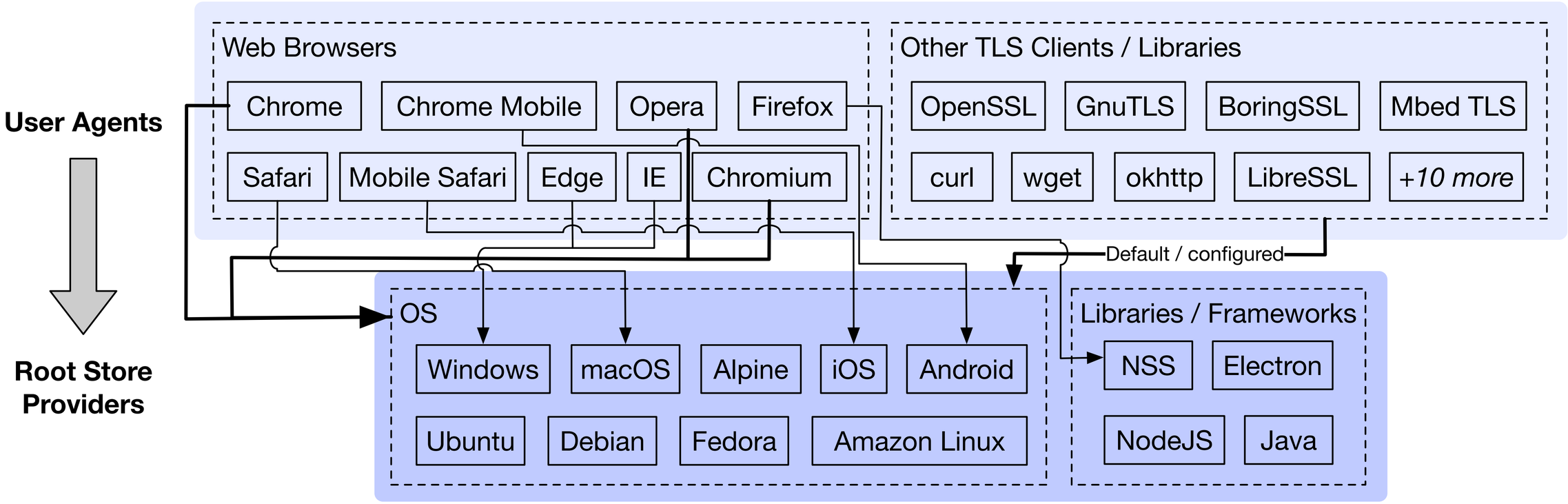
User Agents



Collected root stores for 77% of global CDN top 200 user agents

Determined default root store for dozens of libraries / TLS clients

Root store providers



Research Questions

1. Which root store providers do TLS user agents rely on?

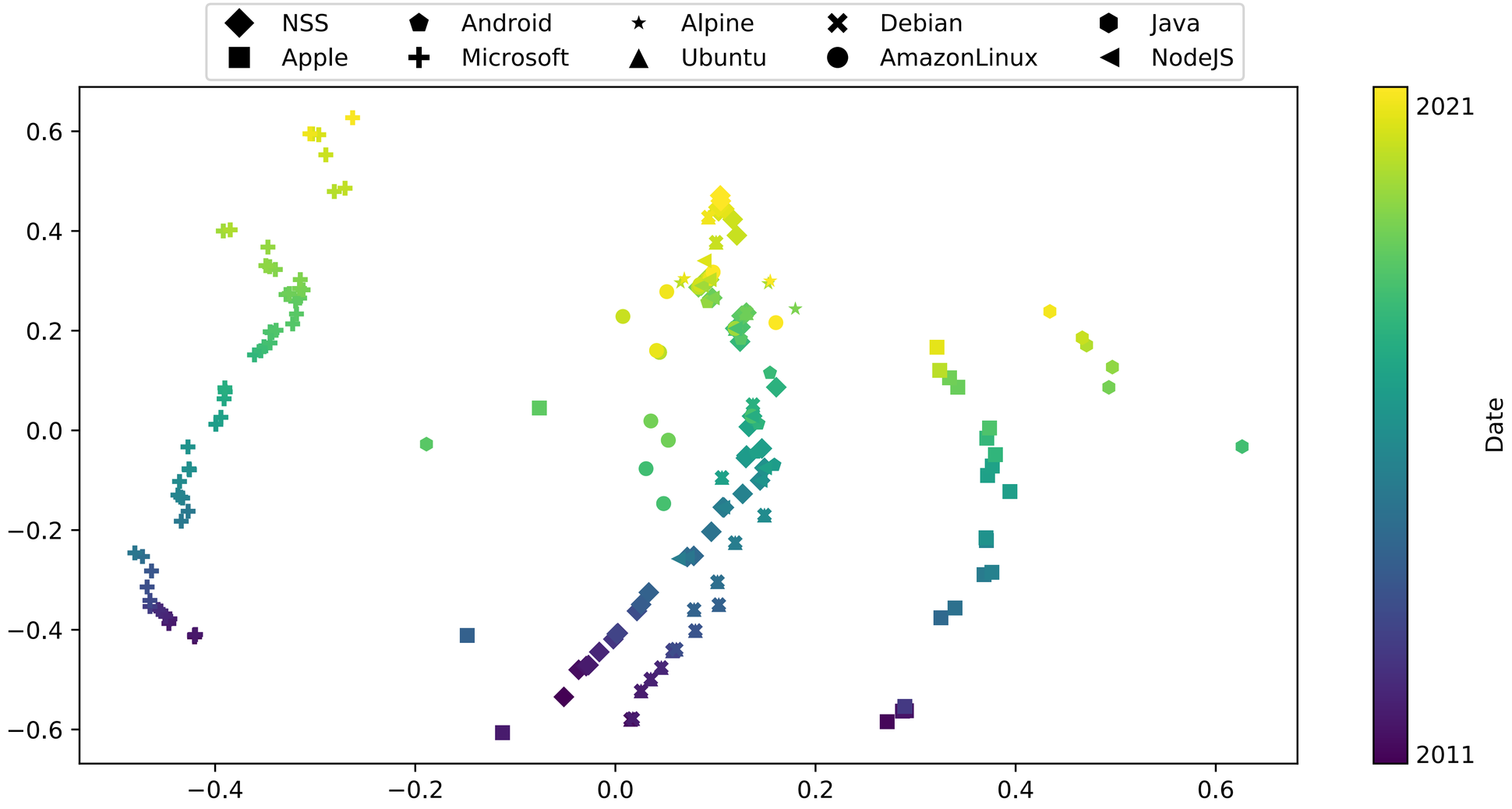
2. How do root store providers determine which CAs to trust?

3. Characterization of root store programs

4. How faithfully do providers copy root program trust?

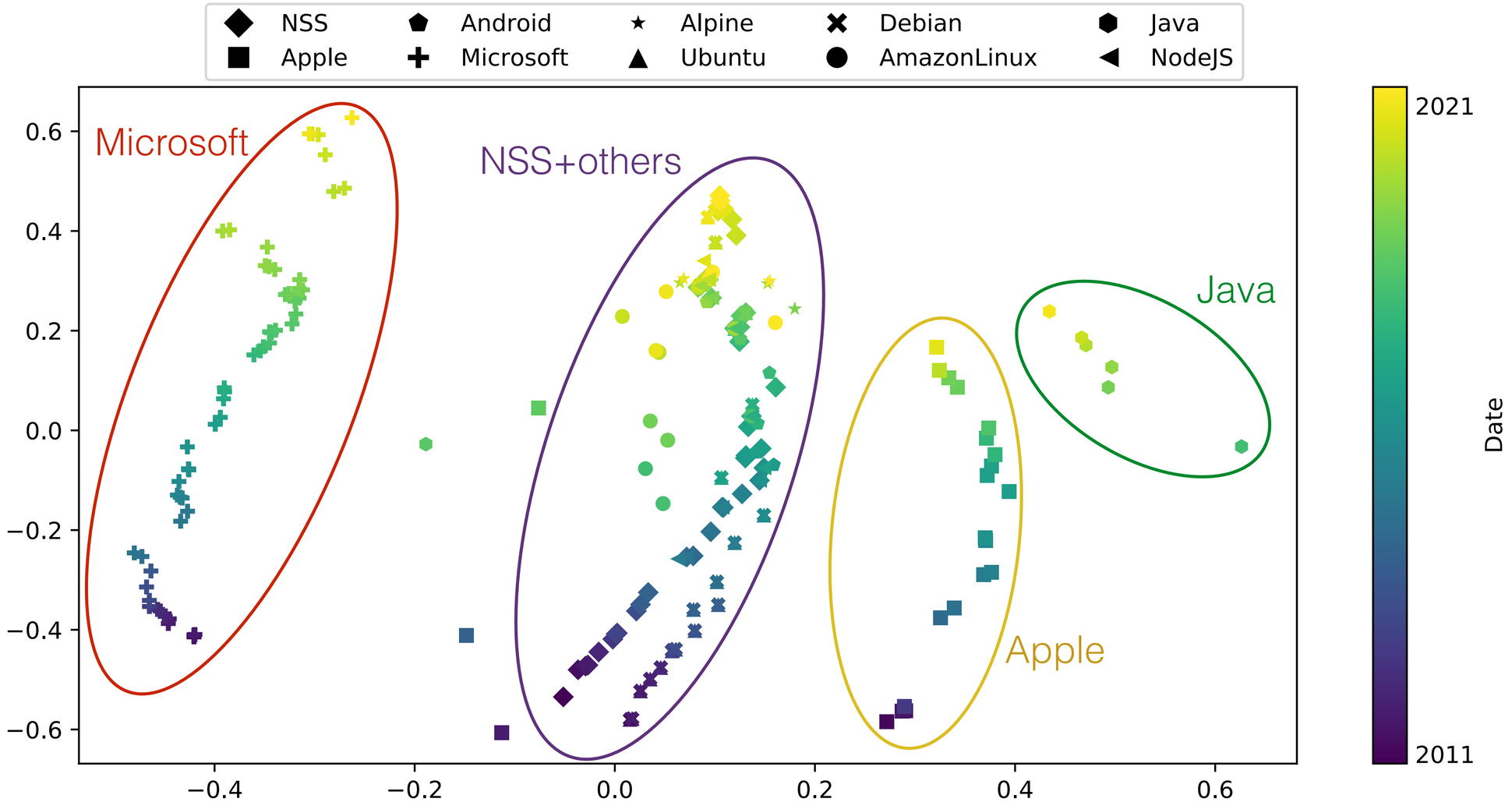
Clustering providers

metric MDS

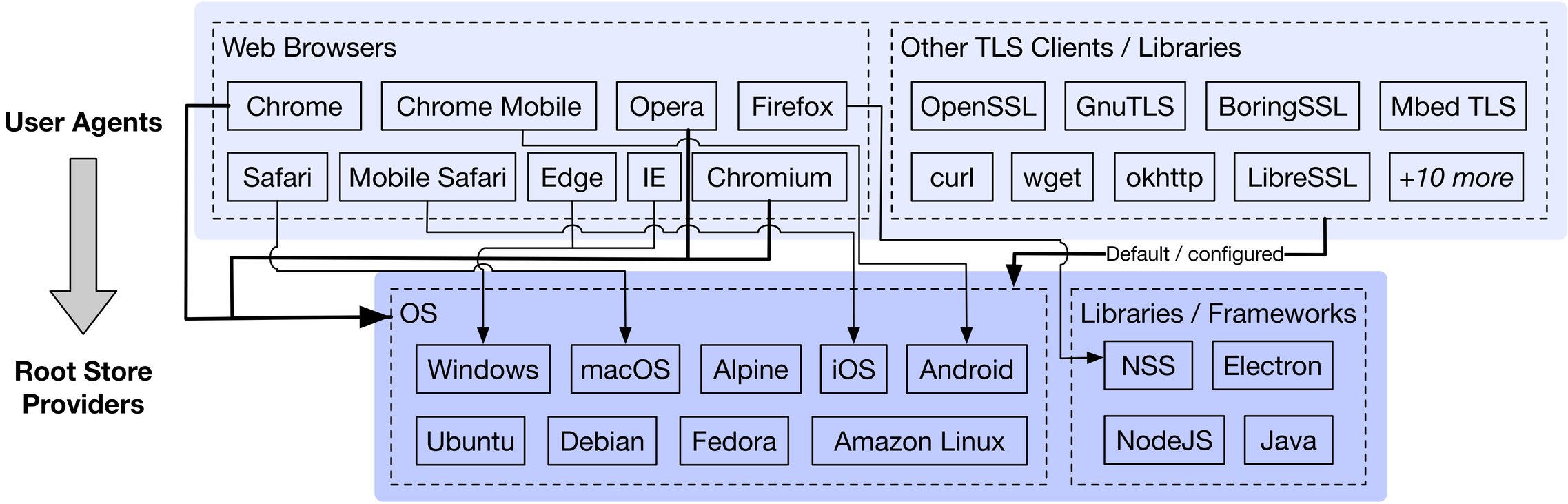


Clustering providers

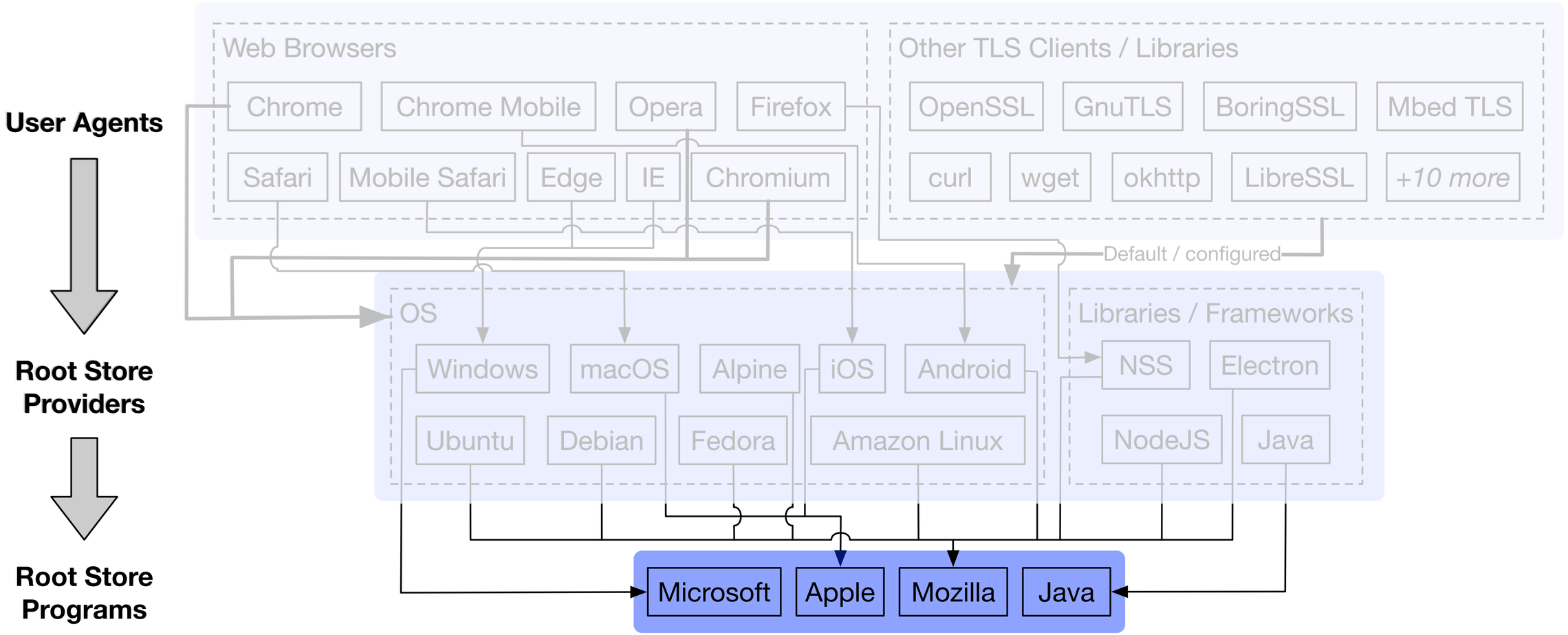
metric MDS



Root store providers



Root store programs



Research Questions

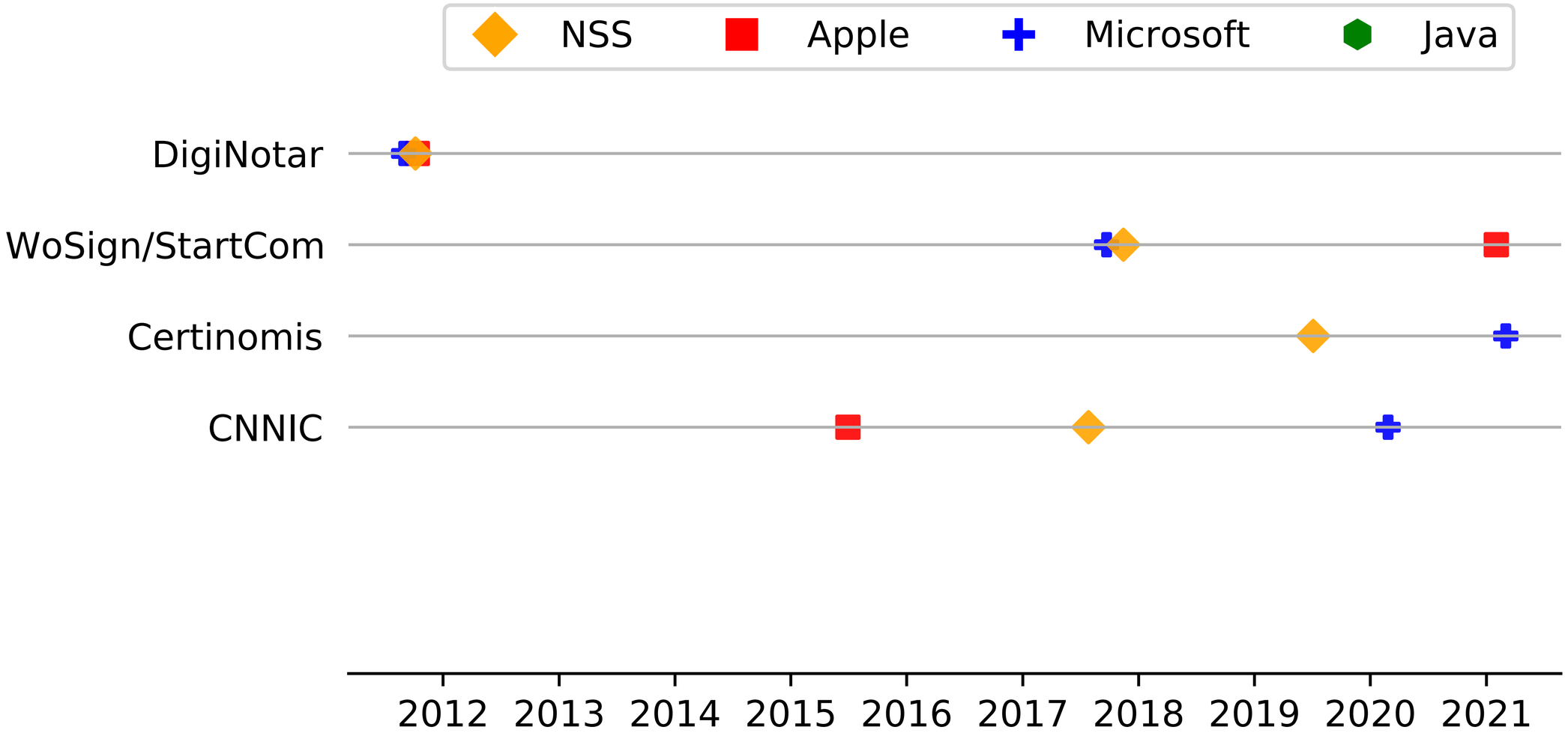
1. Which root store providers do TLS user agents rely on?

2. How do root store providers determine which CAs to trust?

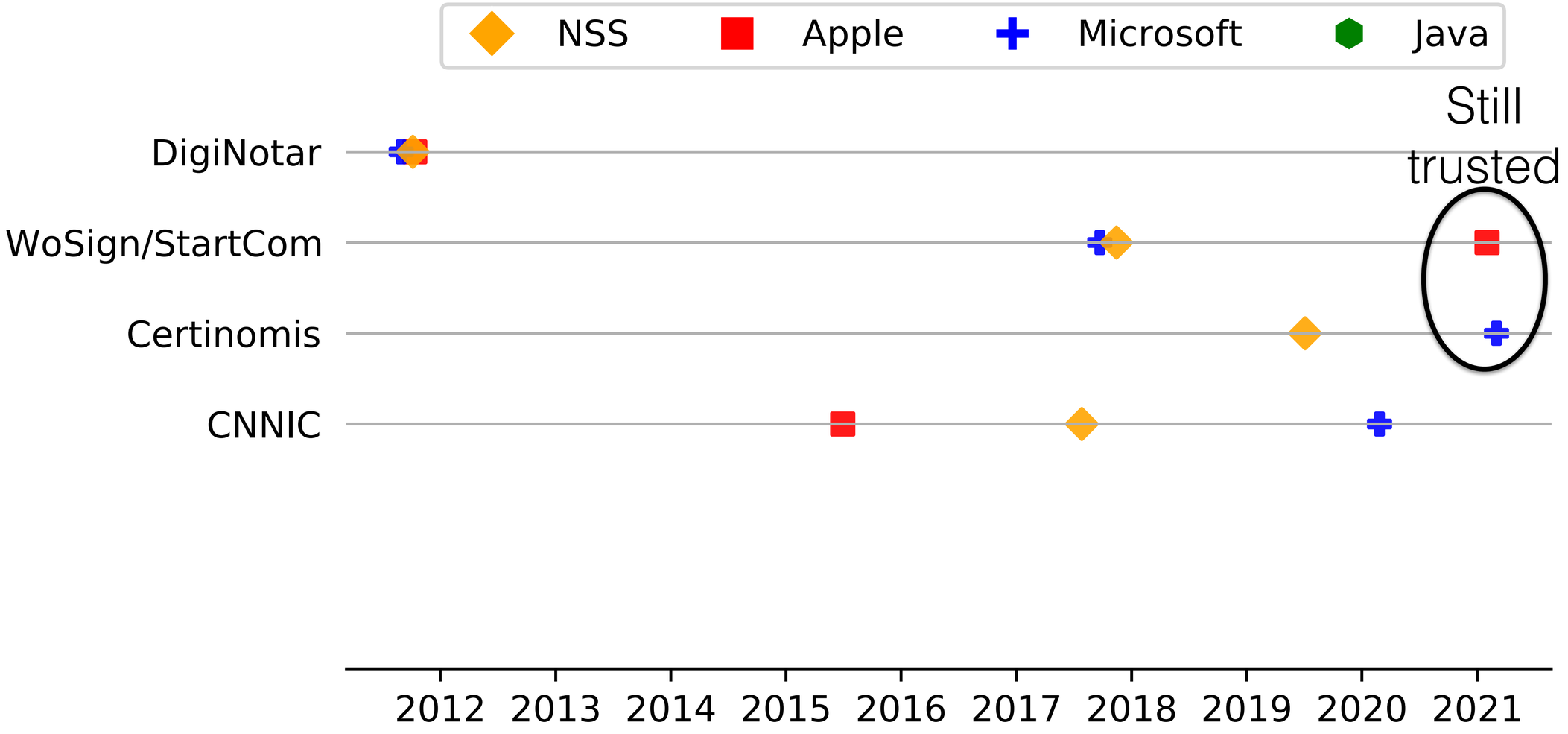
3. Characterization of root store programs

4. How faithfully do providers copy root program trust?

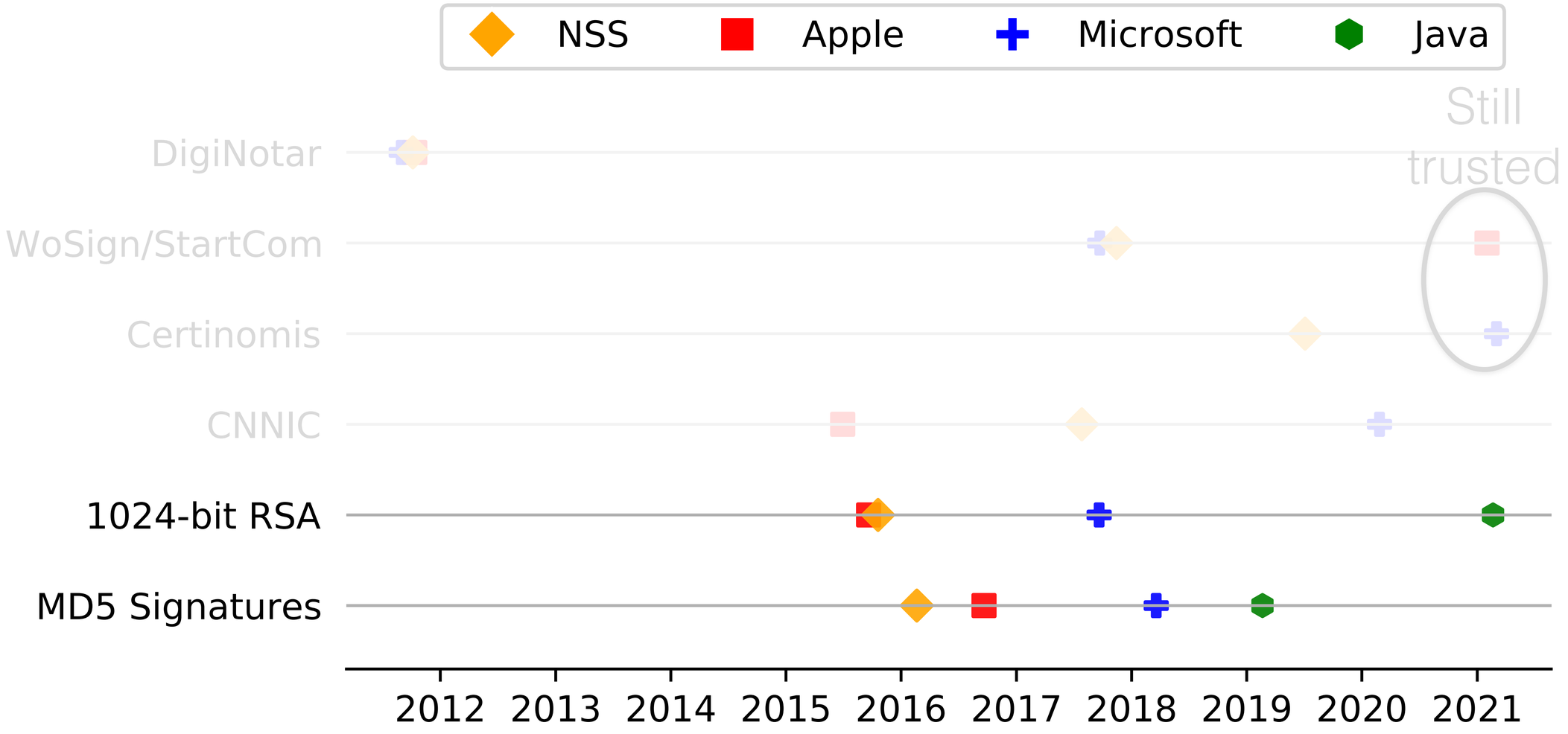
Root program comparison



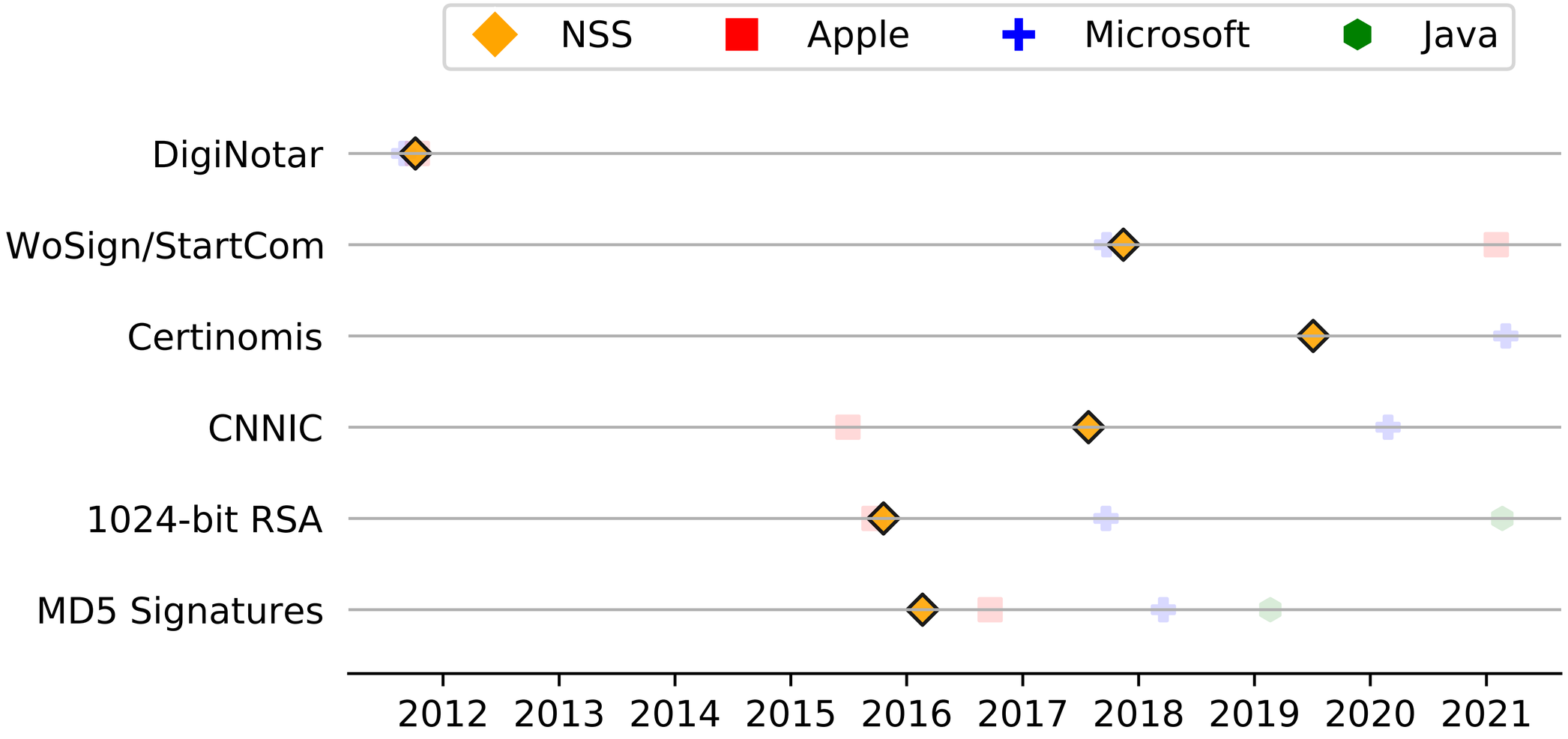
Root program comparison



Root program comparison



Root program comparison



Root program comparison

1. Mozilla responds quickly to CA distrust incidents; Microsoft relatively slow, Apple varies.

2. Apple/Mozilla operate relatively hygienic root stores.

Root program comparison

1. Mozilla responds quickly to CA distrust incidents; Microsoft relatively slow, Apple varies.

2. Apple/Mozilla operate relatively hygienic root stores.

3. Size: Mozilla < Apple < Microsoft; Mozilla most restrictive, Microsoft allows government super-CAs.

Root program comparison

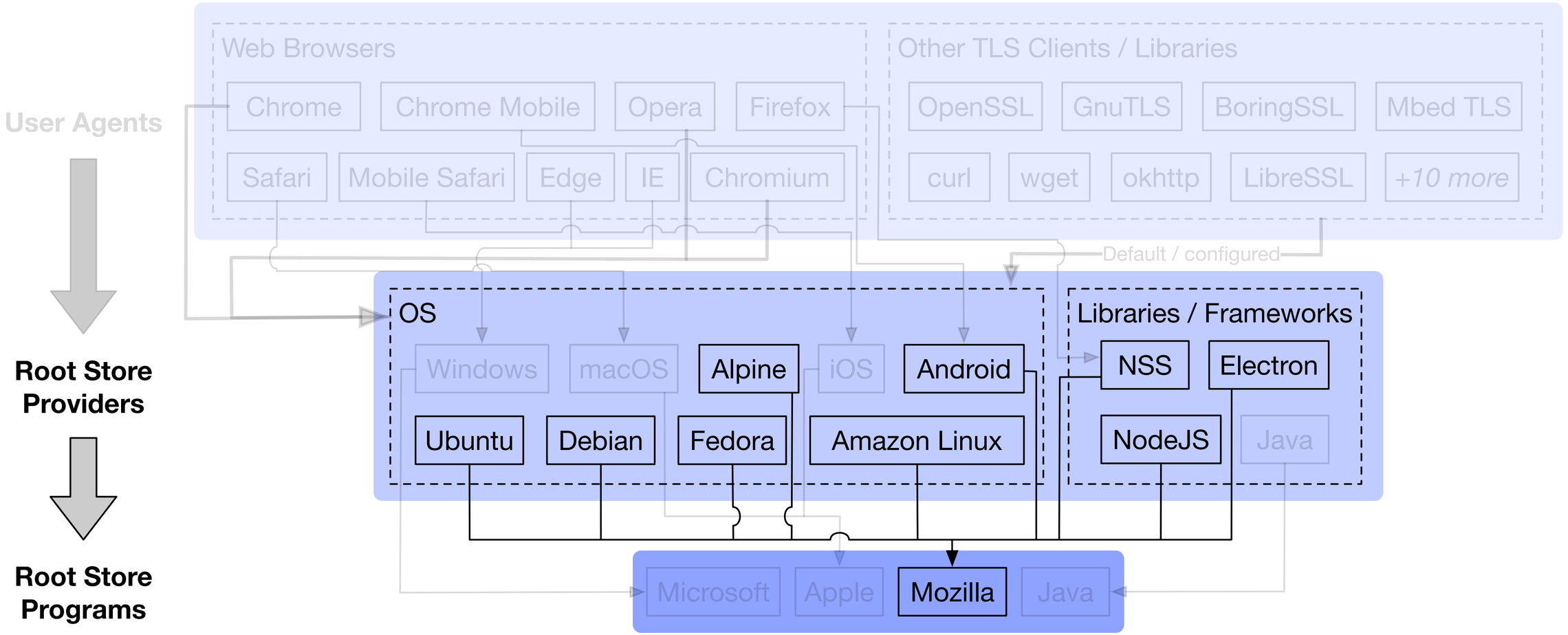
1. Mozilla responds quickly to CA distrust incidents; Microsoft relatively slow, Apple varies.

2. Apple/Mozilla operate relatively hygienic root stores.

3. Size: Mozilla < Apple < Microsoft; Mozilla most restrictive, Microsoft allows government super-CAs.

4. Mozilla runs the most transparent root store program.

Mozilla derivatives



Research Questions

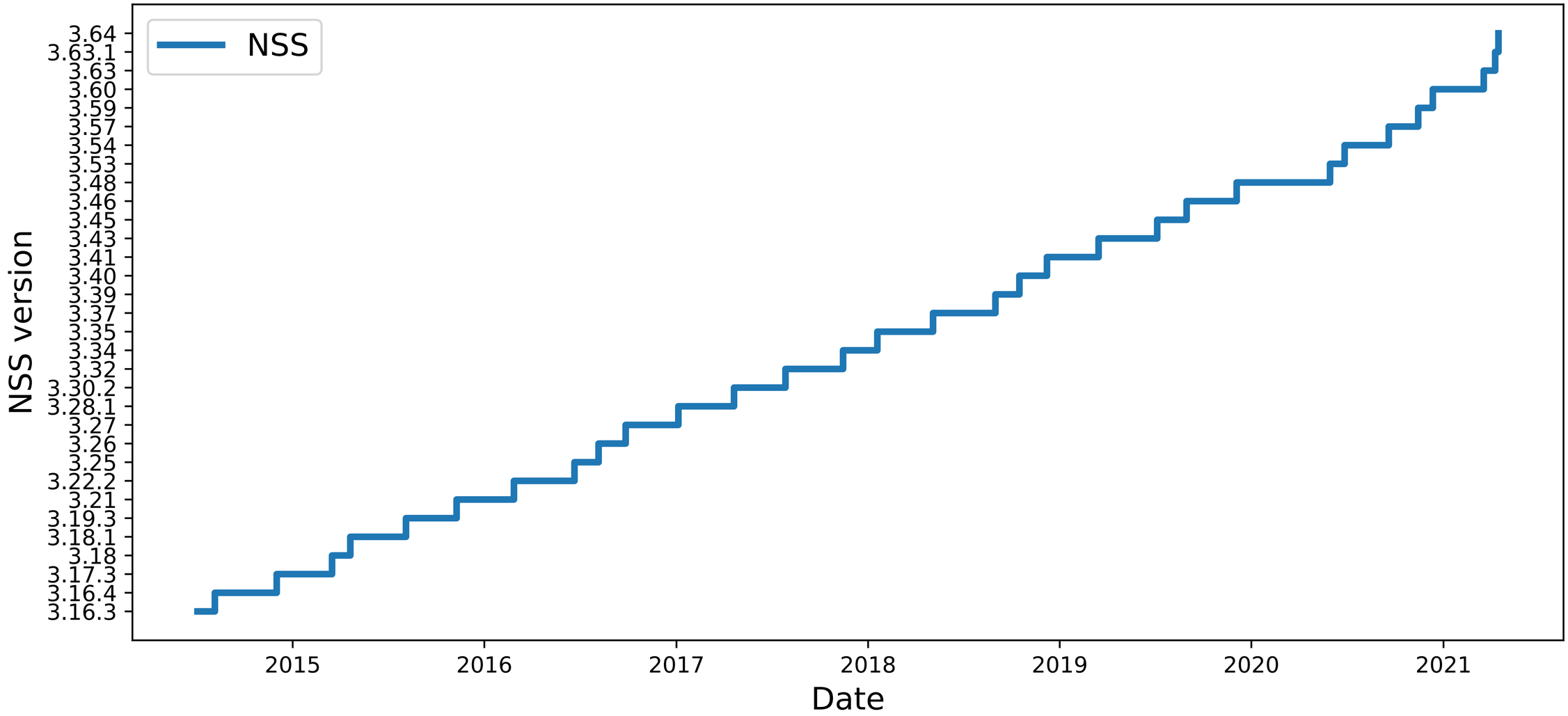
1. Which root store providers do TLS user agents rely on?

2. How do root store providers determine which CAs to trust?

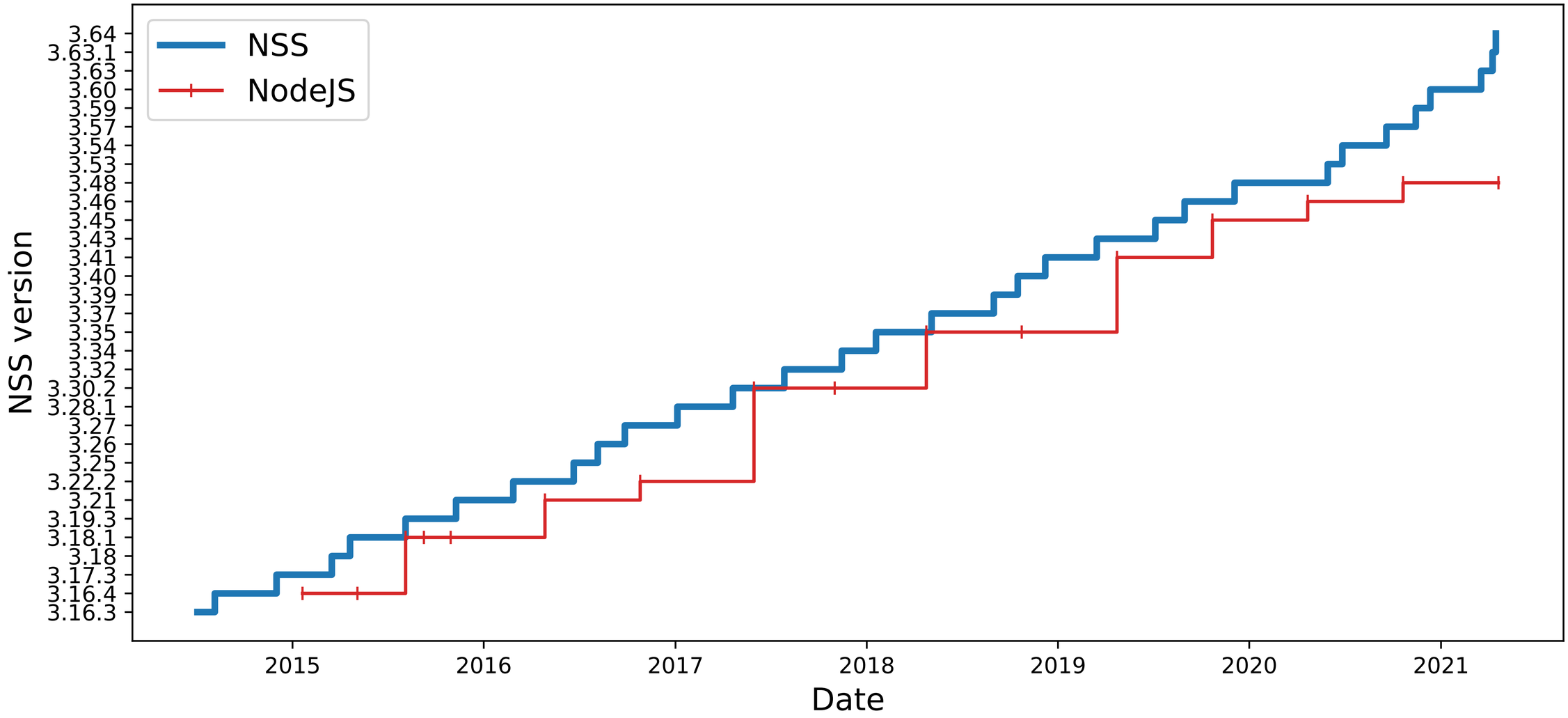
3. Characterization of root store programs

4. How faithfully do providers copy root program trust?

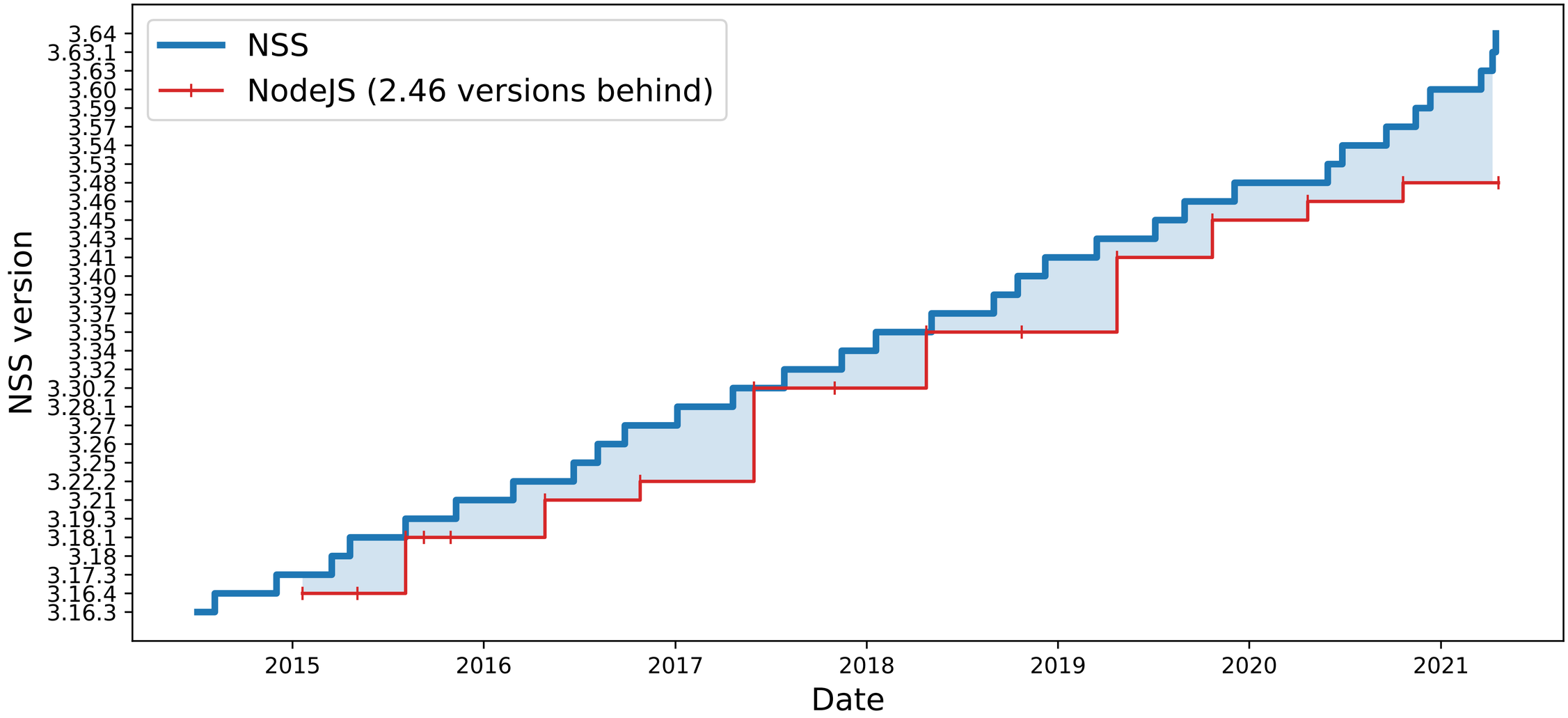
Derivative delay



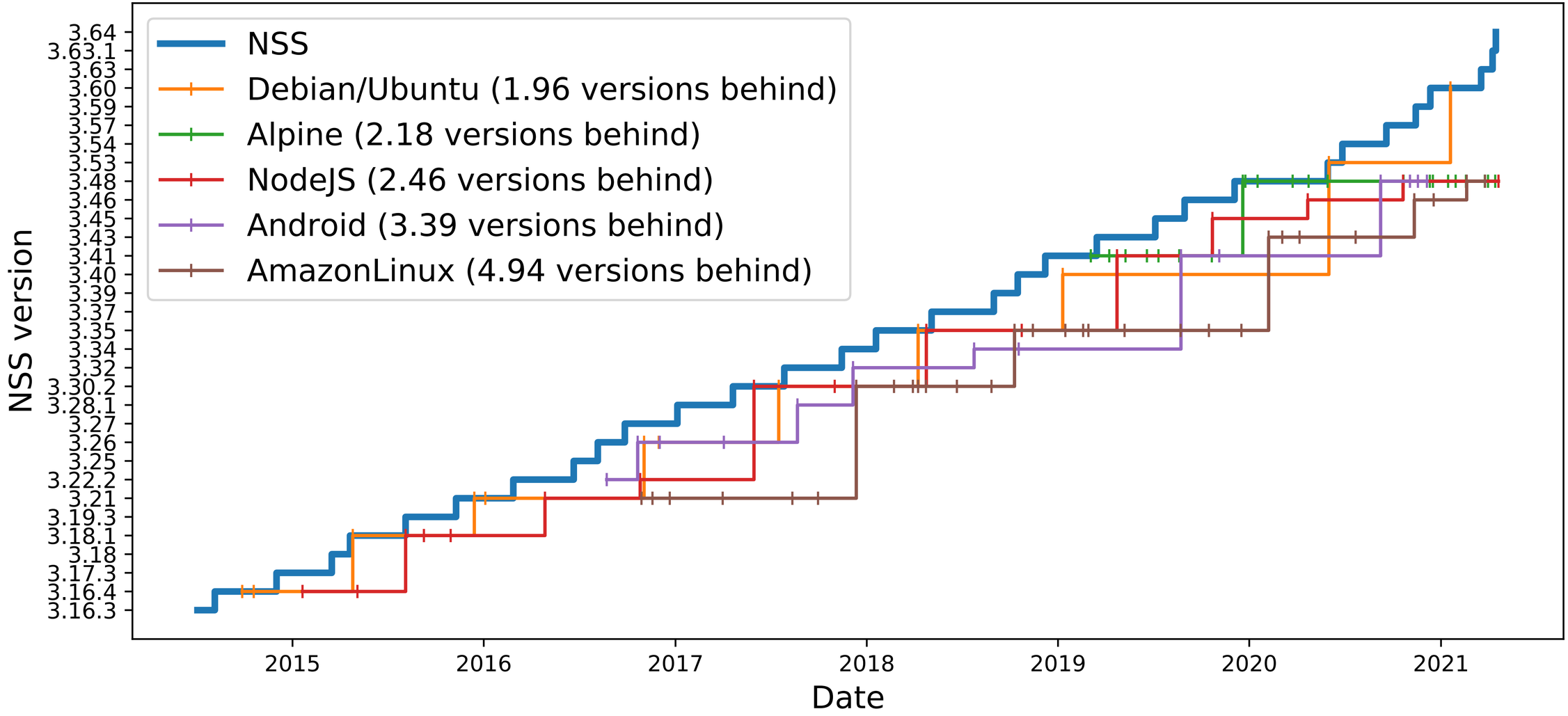
Derivative delay



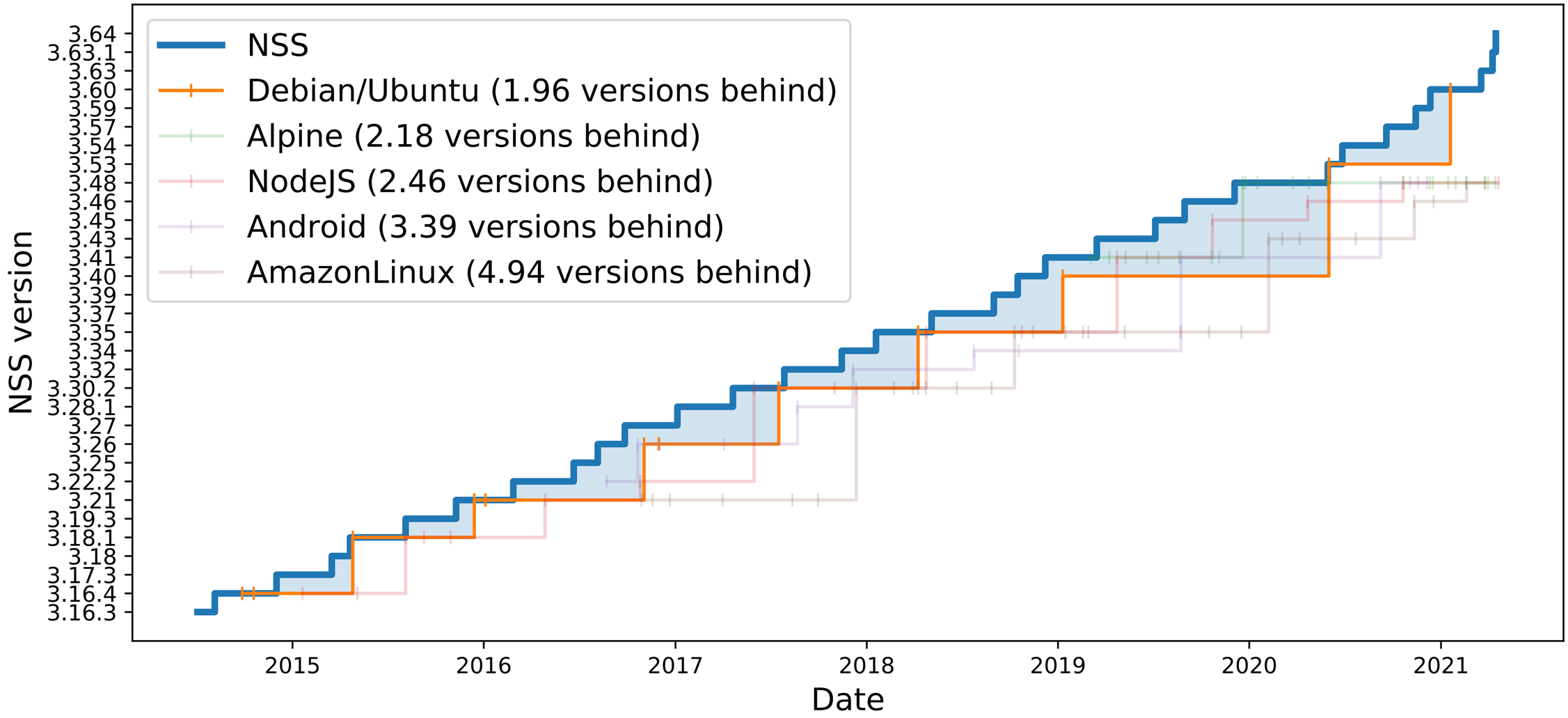
Derivative delay



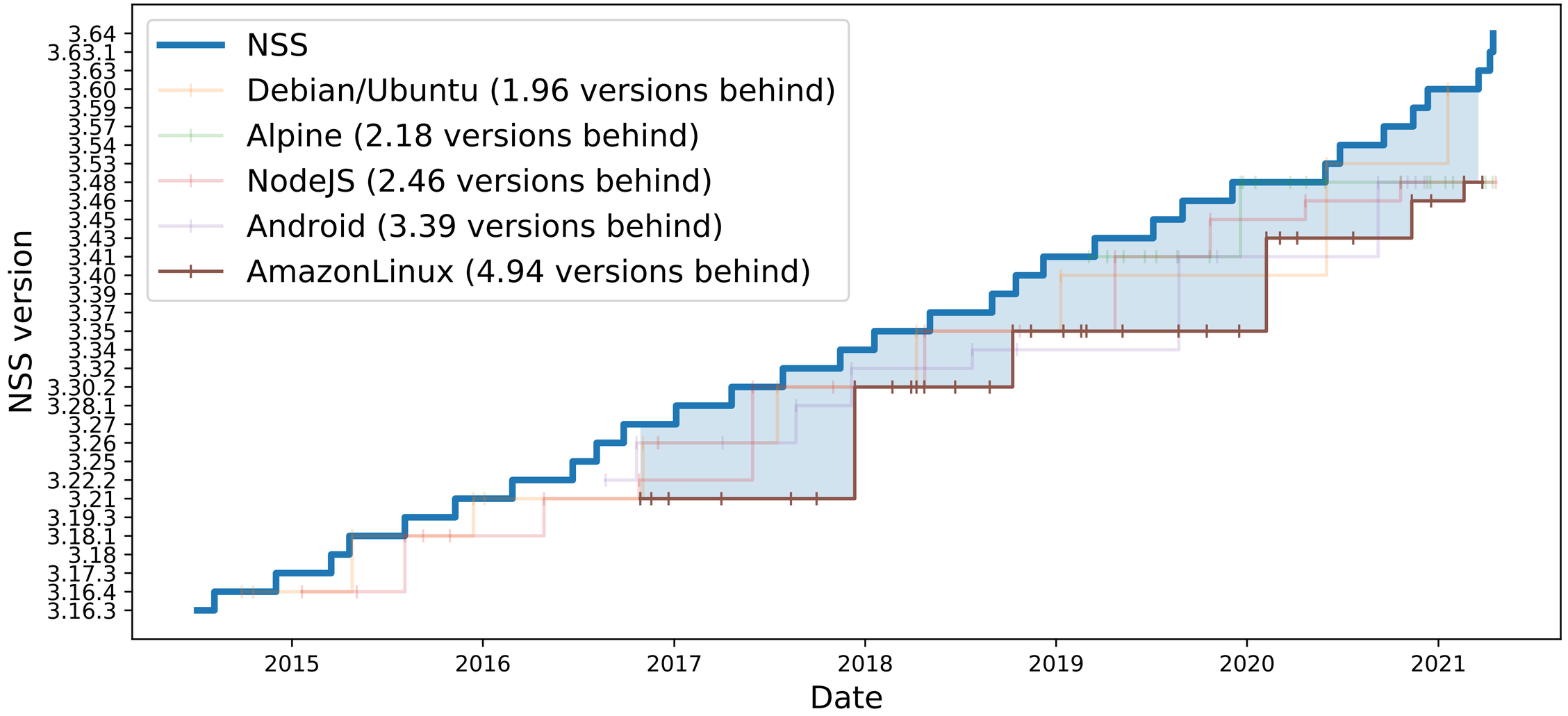
Derivative delay



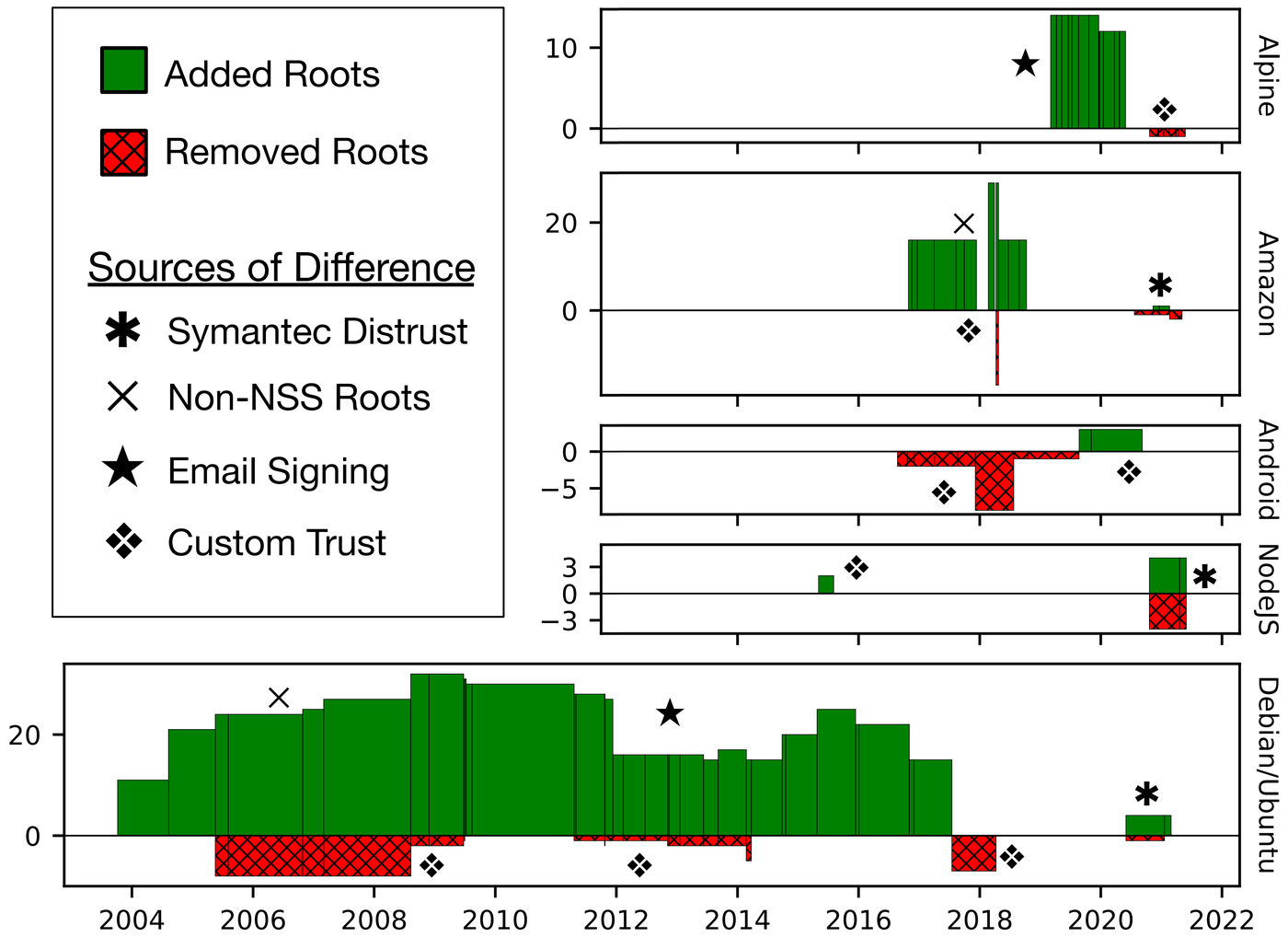
Derivative delay



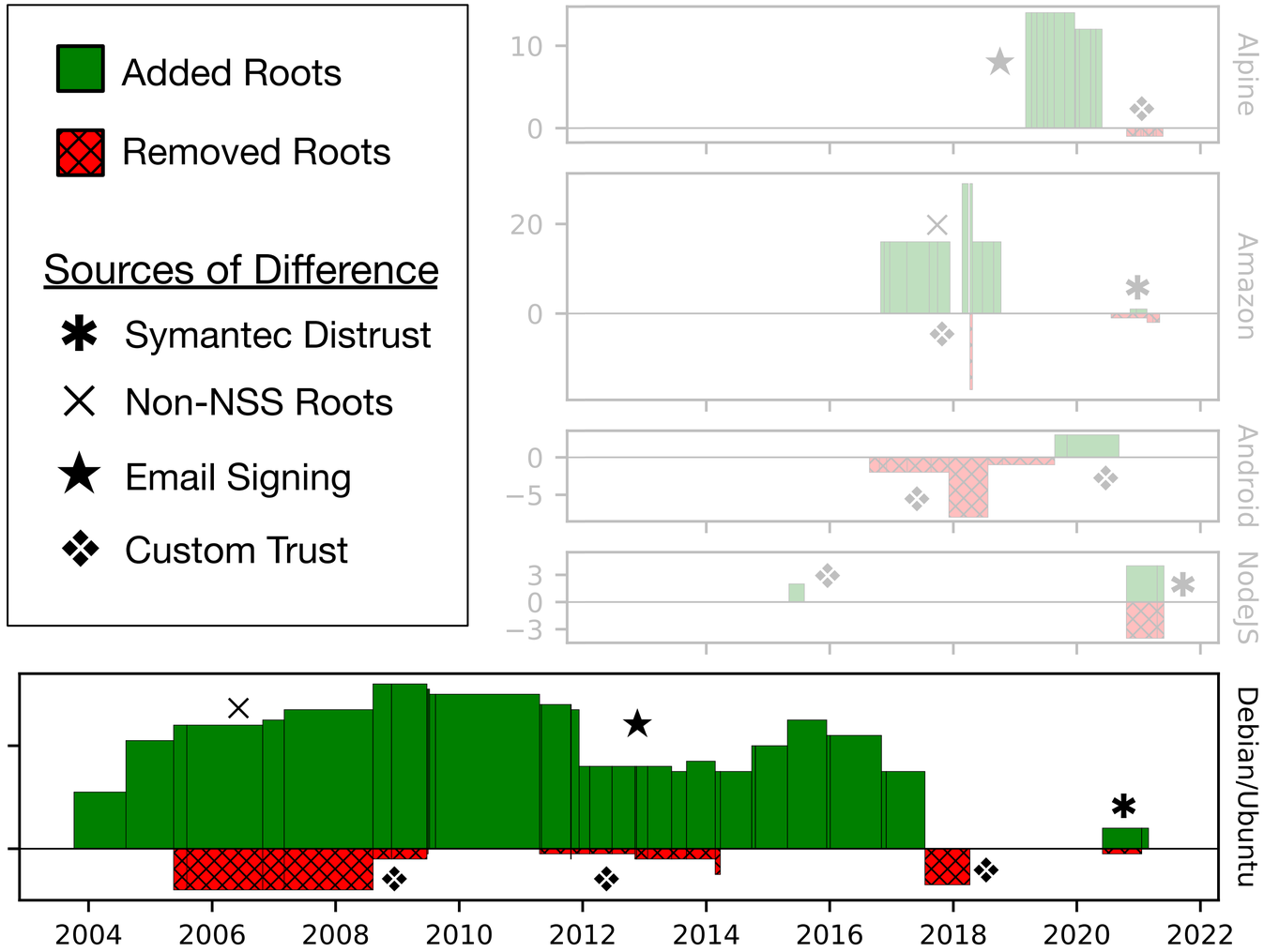
Derivative delay



Trust deviations



Trust deviations



Trust purpose conflation: trusting non-TLS certificates

Partial trust incapability: Symantec distrust dilemma

Non-NSS trusted CAs: questionable trust

App. developer confusion: trusting CAs for code signing, timestamping

Summary

Popular TLS user agents infrequently make their own TLS trust decisions and rely on the OS.

Apple, Microsoft run major root programs; all other root providers originate from Mozilla's NSS root program.

Summary

Popular TLS user agents infrequently make their own TLS trust decisions and rely on the OS.

Apple, Microsoft run major root programs; all other root providers originate from Mozilla's NSS root program.

NSS derivatives copy poorly: delayed updates, questionable bespoke trust, incompatible trust scope.

Summary

Popular TLS user agents infrequently make their own TLS trust decisions and rely on the OS.

Apple, Microsoft run major root programs; all other root providers originate from Mozilla's NSS root program.

NSS derivatives copy poorly: delayed updates, questionable bespoke trust, incompatible trust scope.

Future TLS applications can avoid the rough edges of existing TLS root trust and adopt more modern root store practices.

Tracing Your Roots: Exploring the TLS Trust Anchor Ecosystem

Zane Ma

Postdoc Researcher
Georgia Institute of Technology
zanema@gatech.edu
https://zanema.com