Time: What happens if the world spins backwards?

Jerry Xu
Mentors: Prof. Ari Trachtenberg, Trishita Tiwari

Presented at MIT PRIMES October Conference
Sunday, October 20th, 2019
Overview

- Time on digital devices is synced across the internet
  - Protocol used to sync time is insecure
- Demonstrate a man in the middle attack
- Analyze results and explore possible malicious applications
  - Interfere with human interaction
  - Limit machines’ abilities to self-maintain
  - Undermine security
Why is time important?
How do we sync time?

- Network Time Protocol (NTP; 1985)
- Fundamental Internet Protocol
- Operates on UDP
  - Fast, not reliable
- Designed for precision
How do we sync time?

Level = Stratum (pl. strata)
How do we sync time? – Latency mitigation

Tardy Alice

\[ T_A : \text{Leaves client} \]
How do we sync time? – Latency mitigation

Tardy Alice

$T_A$: Leaves client

NTP Server

$T_B$: Arrives server

$T_C$: Leaves server

Overview

Background

Vulnerability and Exploit

Effects

Potential Resolutions

Boston University Electrical and Computer Engineering
How do we sync time? – Latency mitigation

Overview

Background

Vulnerability and Exploit

Effects

Potential Resolutions
How do we sync time? – Latency mitigation

$T_A$: Leaves client → Latency → $T_B$: Arrives server

$T_D$: Arrives client ← Latency ← $T_C$: Leaves server
How do we sync time? – Latency mitigation

\[ T_B - T_A = \text{offset} + \text{latency} \]
\[ T_D - T_C = (-\text{offset}) + \text{latency} \]

\[ \text{offset} = \frac{(T_b-T_a)-(T_d-T_c)}{2} \]
NTP “Safeguards” in Packet Structure

- Authentication field
- Panic threshold
- Checksum

Conclusion → insecure as consequence of design

Can we modify a packet?  
What will happen as a result?
Exploiting NTP

- Spoofing legitimate NTP server
  - Hard; limited scope
- Modifying packets in transport
  - Easier
  - Active vs passive
    - Active: requires access between target and NTP server
    - “On-path”
    - Passive: no direct access
    - “Off-path”
How do we sync time? – On-path attack

Tardy Alice

$T_A$: Leaves client
How do we sync time? – On-path attack

Tardy Alice

\( T_A : \text{Leaves client} \)

NTP Server

\( T_B : \text{Arrives server} \)

\( T_C : \text{Leaves server} \)
How do we sync time? – On-path attack

Adversarial

Tardy Alice

\( T_A \): Leaves client

Bob

NTP Server

NTP PKT

\( T_B \): PKT S/Arr.

\( T_C \): PKT S/Dep.

Changed \( T_B \)

Changed \( T_C \)

Overview

Background

Vulnerability and Exploit

Effects

Potential Resolutions
How do we sync time? – On-path attack

Tardy Alice

\( T_A \): Leaves client

\( T_D \): Arrives client

NTP Server

\( T_B \): Changed

\( T_C \): Changed

Overview

Background

Vulnerability and Exploit

Effects

Potential Resolutions

*Boston University* Electrical and Computer Engineering
Real-world setup

Tardy Alice

Adversarial Bob

Overview

Background

Vulnerability and Exploit

Effects

Potential Resolutions
Real-world setup – what does Bob see?
Real-world setup – what does Bob see?
Real-world setup – what does Bob see?
Real-world setup – what does Bob see?
Real-world setup – what does Bob see?
Types of modifications – what can Bob do?

- **Direct** – a precise time
  - Difficult to implement; needs guessing at latency
- **Offset** – a fixed deviation from the correct time
  - Easier to implement, but less useful
Platforms Affected

- Linux
- Android
- Apple
- Windows
Effects of changing time

- Superficial changes
  - No data changed; only user-facing GUI

- Noncritical changes
  - Insensitive data changed

- Critical/Theoretical issues
  - Forcing computer to perform detrimental actions
  - Sensitive data changed
Effects of changing time

- **Superficial changes**
  - No data changed; only user-facing GUI

- **Noncritical changes**
  - Insensitive data changed

- **Critical/Theoretical issues**
  - Forcing computer to perform detrimental actions
  - Sensitive data changed
Superficial

- Social media
  - Time-centric
- Does not change actual data
  - Comparison of local time to server
- Graceful handling
  - Use absolute time
  - Use pre-existing timezone strategy
  - Calculate times off-device
Noncritical

- Injection of user-facing data
  - Incorrect sorting by time
  - Change critical metadata
  - Insertion of data where desired

- Invalidating SSL
  - Annoyance to user
  - Kill Email Sync
Critical Issues

- Logging/scheduling (Linux)
  - Cronjobs
    - Scheduled tasks
    - Rely on system time
  - Logging
    - Rotating logs
    - Keeping logs forever
    - Premature removal
  - Multithreaded applications
    - Scheduling tasks
    - Interrupt functionality

Theoretical Issues

- Manipulating SSL
  - Reusing expired certificates
  - HTTP Downgrading
- If direct-time shifting
  - Predicting pseudorandom number generation
Shortcomings/limitations of threat model

- Windows is not exploitable by default
- Needs man in the middle access
  - Limited scope of targets
- Precise time shifting
  - Extremely unreliable
- “Helper” attack
  - Real consequences come when used in conjunction w/ other attacks
Resolving this issue

▪ Fix needs to start with developers of apps and OSes
  ▪ Keep time calculations server-side
  ▪ Use a “time zone” system like iMessage

▪ Re-implementing time sync
  ▪ Use secondary, harder to spoof services: GPS, cell
    ▪ Still vulnerable in general to nation-state attackers
  ▪ Expanding Windows-like authentication system to other platforms
Conclusions/Future Work

- Fundamental protocol’s inherent flaw will be exploited
- Scope of attack is limited but significant
- Big issue: human loss of trust in tech

- Work on implementations of higher-level trust-based attacks
- Target more IoT devices
- Implement security or replace NTP
Special thanks to:

- Prof. Ari Trachtenberg and Trishita Tiwari
- Dr. Aanchal Malhotra
- Prof. Mayank Varia
- My parents
- MIT PRIMES
Any questions?
Academic Credits

Image Credits

- Moon animation: https://media.giphy.com/media/Qllf7zcBVJuak/giphy.gif
- FedEx plane: https://3acujq5da9i3we40i1od3k1/wp-content/uploads/2018/06/fedx_freighter_order3_960x600-696x435.jpg
- CPU clock: https://hsto.org/getpro/habr/post_images/9d4/ede/bb8/9d4edebb8a0253cb1b973bd5df46a9a9.jpg
- SSL certificates: https://www.iconsdb.com/icons/preview/green/ssl-badge-2-xxl.png
- Y2K: https://i.ytimg.com/vi/Q85jernwBc4/maxresdefault.jpg
- GPS: https://www.geotab.com/geomages/blog/what-is-gps.png
- Lock Breaking: http://4.bp.blogspot.com/-Laasnybm00c/TbmmgZTIuiI/AAAAAAAAAC4/uRHCV3CBP3Q/s1600/breakingLock.jpg
- Wave: https://azpng.com/png/2019/06/26/wave-clipart-wifi-waves-blue-transparent-x-free.png
- Tux the Linux Penguin: https://upload.wikimedia.org/wikipedia/commons/a/af/Tux.png
- Android: https://zdnet3.cbsistatic.com/hub/i/2019/08/22/5e05c9d9-27a7-4691-93fa-257717df6582/b96f965a7dee5ea340da1f48eb61a146/android-logo-stacked-rgb.png
- Apple Logo
Additional Information – NTP Packet Structure

<table>
<thead>
<tr>
<th>LI</th>
<th>VN</th>
<th>Mode</th>
<th>Stratum</th>
<th>Poll</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Root Delay</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Root Dispersion</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Reference Identifier</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Reference Timestamp (64)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Origin Timestamp (64)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Receive Timestamp (64)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Transmit Timestamp (64)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional Extension Field 1 (variable)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional Extension Field 2 (variable)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional Key/Algorithm Identifier (32)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional Message Digest (128)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>