

NeoScale Systems Inc.

Hacking FCoE in Absolute Time

(Hacking is used above in the classical, not media, definition)

by Landon Curt Noll

chongo@neoscale.com

For the record: Why do we care?

- NeoScale has no direct product interest in FCoE
 - We officially remain FCoE neutral
- My role as Chief Security Architect includes the identification of crackable elements as well as understanding for impact on security
- As personal note: Historically I have found these keywords are often associated with security concerns:
 - Unbreakable, Secure, Secret, Firewall, Namespace, Password, Key, Strength, Log, Authentication, Authorization, Hash, Random, Entropy, Elliptic Curve, Prime and **Time**

Absolute Time - It is hard!

- *“If you think you understand how to synchronize clocks, you probably don’t. If you know you don’t know all there is (to know) about synchronization, then you probably know more than those who think they do.”*
 - Landon Curt Noll - Summer Usenix 1993
- We do not have a secure protocol to sync clocks
 - NTP cannot withstand determined attacks
 - *SNTP is even worse; whereas timed and friends are “right out!”*
- High availability distributed clock sync is very hard
 - Syncing to Absolute Time under adverse conditions is hard
- Ultimately time is always relative
 - Physics (not standards bodies) is the ruling authority here

Why is establishing time a complex problem?

- You must, at some level, address:
 - Time scales and formats
 - Choice of time reference
 - Reference time calibration methods & calibration errors
 - Time measurement precision & measurement errors
 - Clock rate & variation of clock rate
 - Measurement signal time of flight & variation of time of flight
 - Rate of time & variations of rate of time of flight
- Just to name a few issues ...

Discarding exotic conditions: Why is time hard?

- Network based absolute sync must take into account:
 - Availability of calibrated time
 - Reliability of calibrated time
 - Time of propagation of time sync signals
 - Variation of propagation of time sync signals
 - Detected of false tickers
 - Robustness in the face of false tickers
 - Stability in the face of time quorum flights
 - Time stability sync on a partitioned network
 - Audit logs for time sync events
 - Testing the ability of a time sync network to handle stressful and/or failure conditions

But I use NTP, doesn't that hide time complexity?

- NTP code is subtle and complex
- Too many appliances have fragile / buggy NTP service
- NTP stratum networks rely on external sources
 - Many stratum 1 sources are really stratum 2
 - True stratum 1 sources are expensive & high maintenance
- Clock peer network design is not straight forward
 - Must handle various network partitions problems
 - Must handle network congestion / latency problems
 - Must avoid over-peering and time quorum fights
 - Must handle false tickers and low stratum false tickers
- SNTP is even less robust and less secure than NTP!

The complexity of time sync is significant

- *“The complexity of the physics & security of time (has an order of complexity) similar to that of cryptography”*
 - Landon Curt Noll - Summer Usenix 1993

2x2 Source Threat Matrix

- Where do security threats come from?

<i>2x2 Source</i>	Inside	Outside
Passive	<i>Inside Passive</i>	<i>Outside Passive</i>
Active	<i>Inside Active</i>	<i>Outside Active</i>

- Security covers: Reliability, Availability & Integrity

Why firewalls may not stop external attacks

- “The continued presence of nodes running Operating Systems and Application platforms that remain vulnerable to virus attacks;
 - because they were designed by people who have a commitment to original design flaws
- The continued presence of network nodes infected viruses and other malware filth;
 - despite the best efforts of Anti-Virus and Anti-spyware
- Makes the distinction between the inside and the outside of a network purely an academic exercise.”

• *Landon Curt Noll - 08 Aug 2007 T11*

Outside/Passive attacks on FCoE timestamps

- A “back-hoe partitioned” WAN is later reconnected
 - A clock fight occurs because clocks in a WAN segment drifted
 - Nodes that drifted into the future may delay FC frames long enough to appear “current”
- Packets from an older FC Exchange may be incorrectly accepted as part of a current FC Exchange

Outside/Active attacks on FCoE timestamps

- Attacker logically or physically partitions the WAN so that:
 - A clock fight occurs because clocks in a WAN segment drifted
 - Nodes that drifted into the future may delay FC frames long enough to appear “current”
- Packets from an older FC Exchange may be incorrectly accepted as part of a current FC Exchange

Inside/Active attacks on FCoE timestamps

- Attacker uses DoS to:
 - congest a node
 - make it easier to forge SNTP packets that adjust the node's clock
- The DoS attack stops. Delayed packets belonging to an old FC Exchange and time-stamped in the future may be accepted as valid in the current FC Exchange.

Inside/Passive attacks on FCoE timestamps

- Poor network design creates conditions where:
 - packets may live for a long time
 - conditions where some clocks can drift into the future
- Packets from an old FC Exchange are incorrectly accepted as valid in a current FC Exchange
- The faith in FCoE timestamps leads to a false sense of safety



"We've considered every potential risk except
the risks of avoiding all risks."

Recommendations

- Do not depend on Absolute Time
 - Use relative Time
 - *Time since the start of some event*
 - Amount of time spent since a FC Frame entered a node
 - Use accumulated Time
 - *Time during some period or periods*
 - Total accumulated time spent by a FC Frame transiting nodes
 - Use countdown counters
 - *“Take one down, pass it around, 98 hops left to go with this frame!”*
 - Do not depend on Absolute Time to protect FC storage