A Devil of a Time: How Vulnerable is NTP to Malicious Timeservers?

Original research by:
Yarin Perry, Neta Rozen-Schiff and Michael Schapira
The Hebrew University of Jerusalem

Presented by:
Lucas Betts

Background

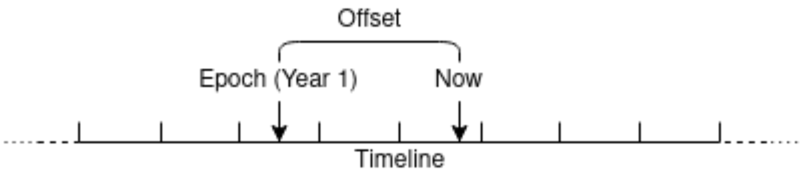
A Brief Introduction to "Time"

As humans, we consider "the time" to be an offset against a fixed point or **Epoch**.

Most of us use the end of 1BC and beginning of 1AD in the gregorian calendar as the epoch and roughly consider 2021 years, 9 months and 23 days to have passed since then.

Alternate calendars exist, such as the Chinese calendar and Jewish/Hebrew calendar.

The transfer of time between people is based on **trust.**



Computer Time

Computers use a different epoch to humans, and have to count in specific increments (each bit in a certain data structure).

Often, the time offset is calculated as seconds and fractions of a second from an epoch.

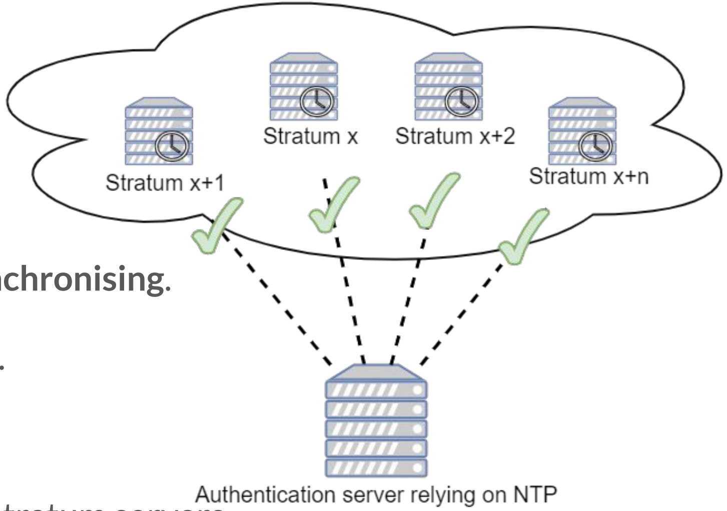
The time cannot be permanently set from the factory, as power cuts and flat batteries would reset this. Additionally, clock drift will gradually cause some machines to lose time.

Need to be able to come to an **accurate** time consensus over **high-latency networks**.

The increasing number of computers requiring accurate times means the proposed solution must be **distributed** and **robust**.

The number of different computer time epochs requires something standardised, which different operating systems can handle in their own client implementations.

Network Time Protocol (NTP)



NTP clients make requests to **NTP timeservers**, which is called **synchronising**.

Timeserver trustworthiness is defined by a number called **Stratum**.

The stratum is the distance from a true-time source (stratum 0).

Timeservers use the **client NTP protocol** to synchronise to lower-stratum servers.

Timeservers of equal stratum are called **peers**, and requests made between is called peering.

NTP uses an epoch of Jan 1 1900 and will rollover on Feb 7 2036

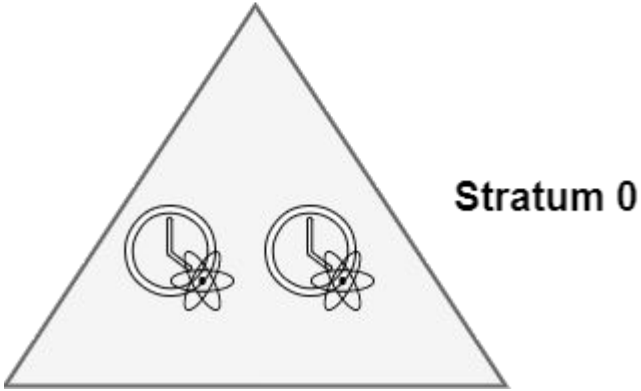
NTP counts in increments of 233 picoseconds

Stratum 0

Atomic clocks or GPS Antennae.

Not connected to the internet (no latency calculation).

Considered extremely accurate.

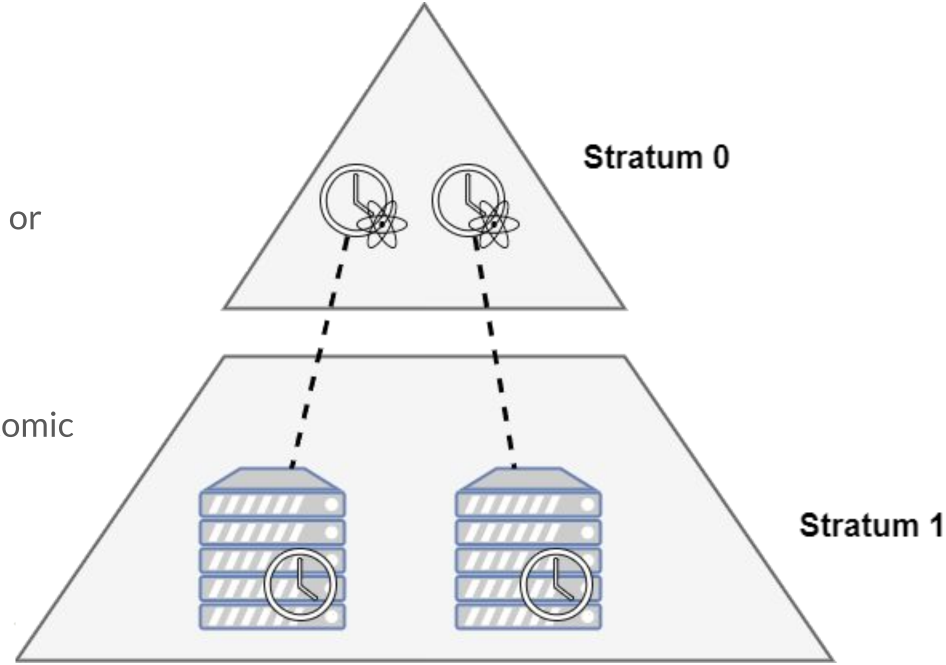


Stratum 1

Synchronises with Stratum 0 devices only over local or internal networks.

Accessible via networks such as the Internet.

Often specialised hardware containing Stratum 0 atomic clocks etc.



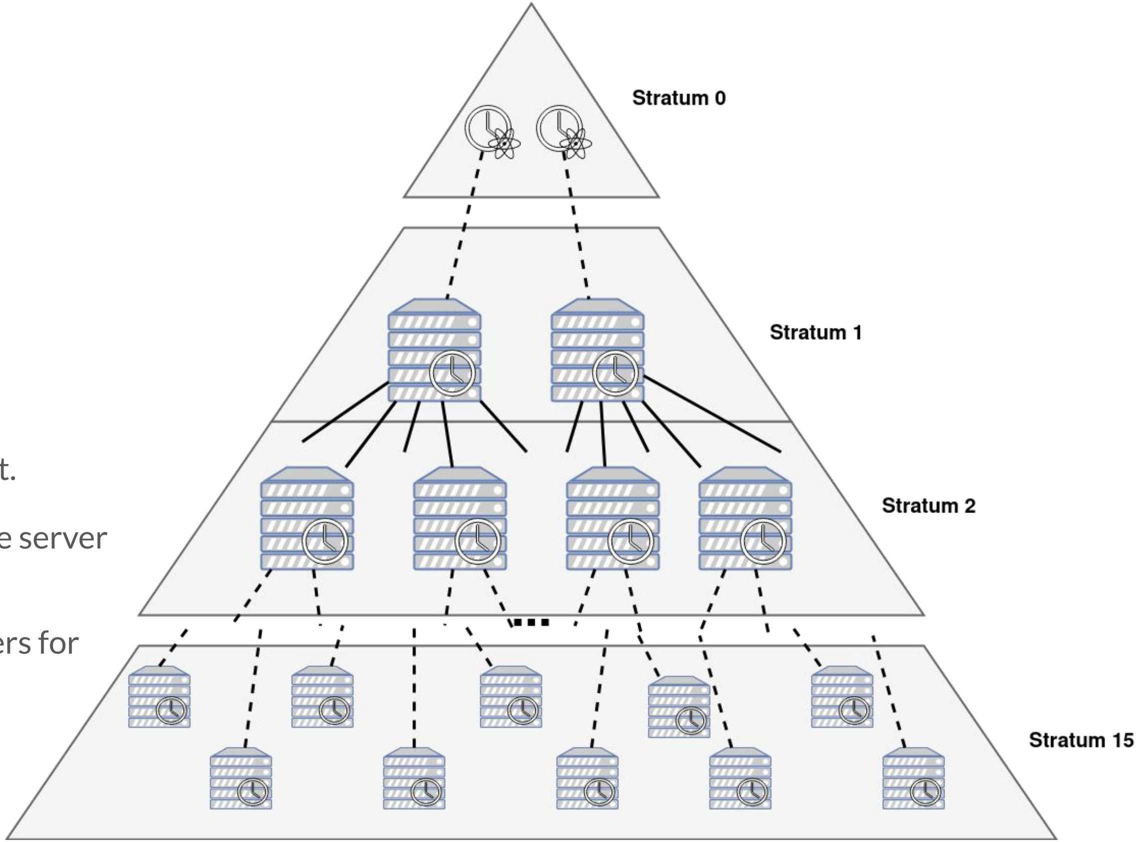
Stratum 1-15

Stratum 2 Synchronises with Stratum 1.
Stratum 3 synchronises with Stratum 2.
...
Stratum 15 synchronises with Stratum 14.

Accessible via networks such as the Internet.

The higher the stratum, the less accurate the server may be.

Servers often peer with same-stratum servers for accuracy and redundancy purposes.



NTP Clients

Clients are either **manually configured** or **dynamically receive** lists of timeservers.

Dynamic allocation comes from **Pools** such as the NTP Pool Project.

Make periodic requests to these timeservers.

Run algorithms to calculate the current time based on:

- Time provided by timeserver(s)
- Network latency calculations on each provided message

How is NTP Vulnerable to attack

Existing communication and cryptographic protocols require a consensus for time at both ends.

By manipulating responses to NTP clients, we are able to change the perceived time at either a particular client or a large demographic of clients.

This is known as a **Timeshift attack.**

Often the attack is to deny accessibility, not to defeat authentication or data integrity.

Other attacks exist surrounding NTP, however this paper only investigates the timeshift attack.

How much time to shift?

Depends on the protocol you want to attack.

TLS: Years

BGP/OSPF (Internet routing): Days

Blockchain: Hours

Client authentication (i.e. Kerberos): Minutes

The Experiment

The researchers considered two attack vectors, namely what servers are malicious.

Use Existing Timeservers	Add New Timeservers
Requires either: 1. Access to an organisation with existing timeservers 2. Unintended access to existing timeservers (e.g. via software vulnerability) 3. Able to masquerade as existing timeservers (e.g. via DNS Cache poisoning or BGP IP Prefix hijacking)	Does not require access to existing hardware. More difficult to affect machines with manually configured timeservers. Need to insert new machines into pools.

Setting up the timeserver.

Trivial to insert a malicious timeserver into a new pool.

Once the actor has access to timeservers, they may configure the device to optimise impact.

Modifying stratum may not significantly impact effectiveness.

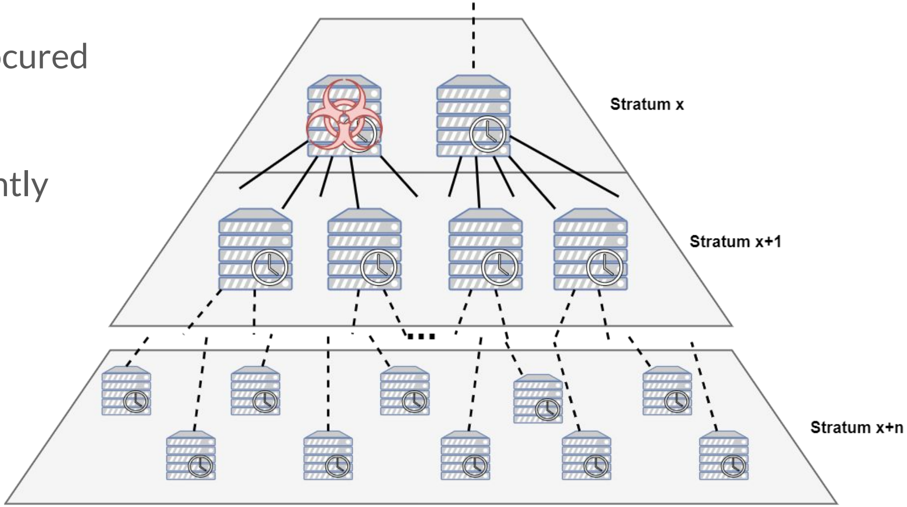
A server-provided parameter **netspeed** was shown to be the most significant.

Providing accurate responses to peers and pool-checks may aid in hiding the malicious actions.

What might this attack look like?

A malicious timeserver is either added to a pool or procured as explained in the previous slide.

The malicious timeserver may selectively or permanently output the incorrect time to other machines.

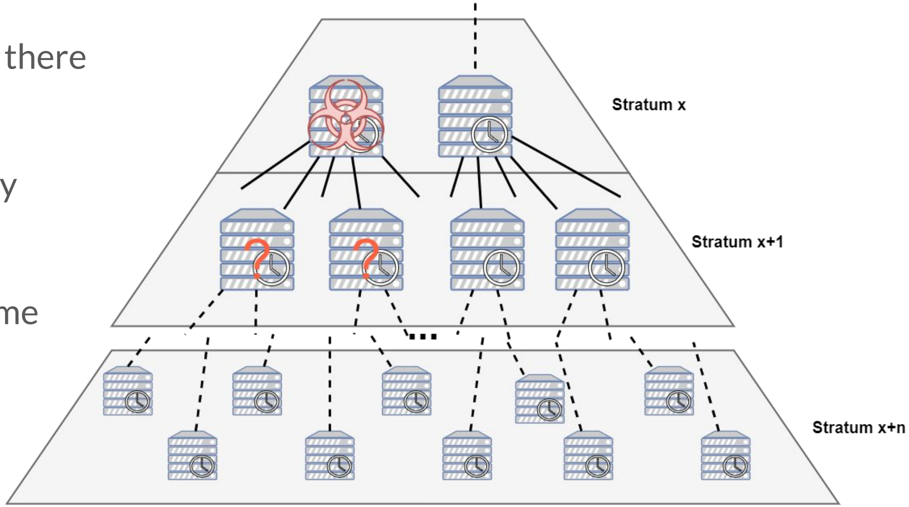


What might this attack look like?

As machines are provided incorrect data, they believe there is a new time.

No hijacking or malicious code is running on them, they believe they are making the correct decision.

They may start responding with the new (incorrect) time when synchronised or peered with.

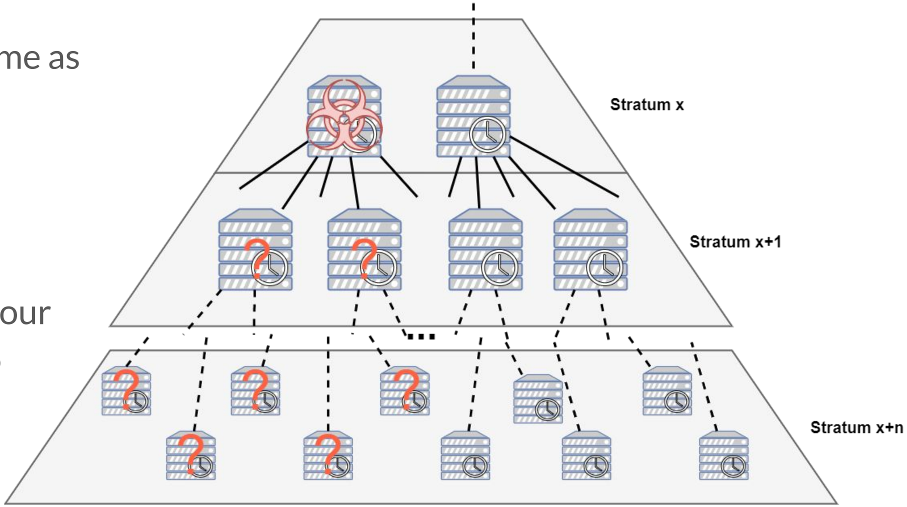


What might this attack look like?

Eventually, more and more timeservers may see the time as being incorrect.

These will all start responding with incorrect times to legitimate synchronisation/peering requests.

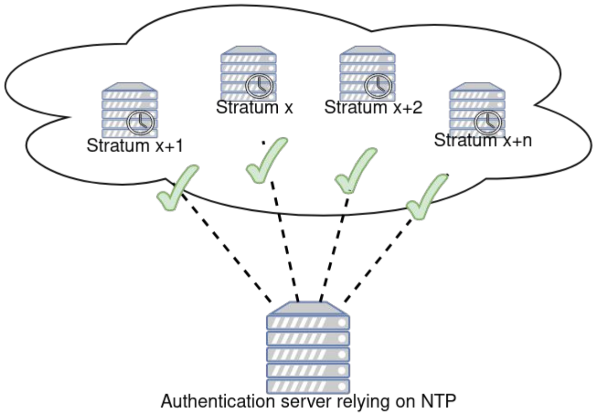
The lower stratum and more synchronising requests your malicious timeserver can get, the more you are able to disrupt time at the target machines.



Effect at the client

As mentioned previously, clients may synchronise with many timeservers for accurate measurements.

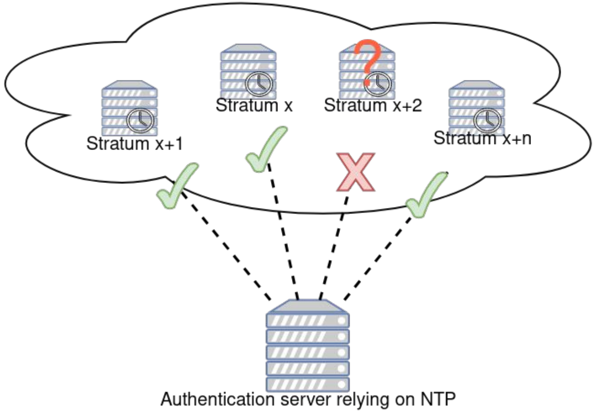
If all timeservers are in consensus, the measurement is accepted.



(Green tick indicates measurement is accepted as correct)
(Red cross indicates measurement is discarded)

Effect at the client

If some timeservers are incorrect after latency adjustments, the client should and will discard that data and use the consensus from the majority.



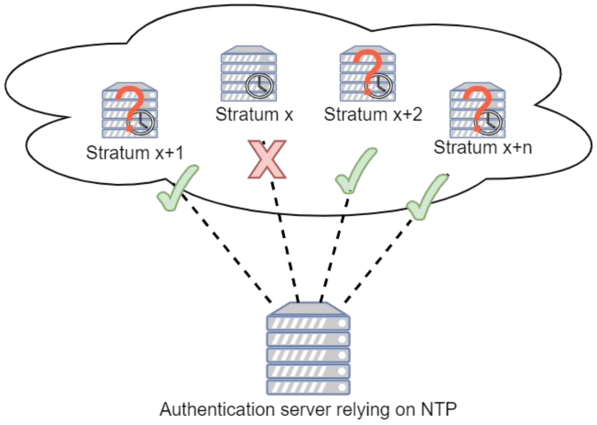
(Green tick indicates measurement is accepted as correct)
(Red cross indicates measurement is discarded)

Effect at the client

However, if the majority of synchronising responses are incorrect, the client will accept the incorrect response and have an equally incorrect sense of time.

This will affect the functionality of many systems that the client may use.

(Green tick indicates measurement is accepted as correct)
(Red cross indicates measurement is discarded)



Research Findings

NTP is vulnerable to opportunistic attackers (disrupt large numbers of client to wreak havoc).

NTP is vulnerable to targeted attackers (disrupt specific or targeted clients for specific purpose).

Adding new timeservers to existing pools may be too easy.

Lying about stratum is not useful.

Lying about network capacity allows orders of magnitude more victims as pools will allocate you to more clients.

Manipulating lower stratum timeservers with many synchronising higher stratum timeservers and clients will have more impact.

Timeservers can still have synchronising requests from clients after being removed from the pool.

Prior Security Approaches

NTP Traffic authentication:

The research divulged that although there are availabilities for NTP to be authenticated, it often is not.

Even with authentication, an attacker that can delay or discard traffic can manipulate perceived time.

Encryption and authentication do not solve attackers that are in control of timeservers.

More robust NTP clients:

Clients such as Chronos use significantly larger number of timeservers for synchronising requests.

Results still show that a motivated attacker can influence enough timeservers to be effective.

Solution

Solution requirements

The authors identified three criteria that a solution must meet:

1. Must preserve the accuracy and precision of today's NTP.
2. Must respect load distribution over timeservers.
3. Must enhance NTP's security against malicious timeservers.

Two Synchronisation Processes

Use existing NTP protocol to determine primary local time.

Secondary synchronisation process called synchronises only with trusted, pool-provided stratum 1 timeservers known as the **Ananke** pool.

If these do not deviate, the primary time is used. If a deviation occurs, an attack is expected to have occured and the watchdog process is used for local time.

Purpose of Ananke

Stratum 1 timeservers do not synchronise with other network-connected timeservers.

Reduces risk of MITM attack between synchronising timeservers.

Ananke pool should not include every stratum 1 timeserver in the pool, only cautiously selected timeservers.

Manual approval of stratum 1 timeservers into the Ananke pool should be used, with only reputable organisations allowed to host these timeservers.

Why not just use the Ananke pool?

Criteria 2 required a solution to respect appropriate load distribution.

Infrequent requests to the Ananke pool, to ensure time does not stray from expected by malicious methods.

Frequent requests to the Primary pool, to reduce potential clock drift at the client.

This satisfies the first criteria of retaining current accuracy and precision.

Solution requirements

The authors identified three criteria that a solution must meet:

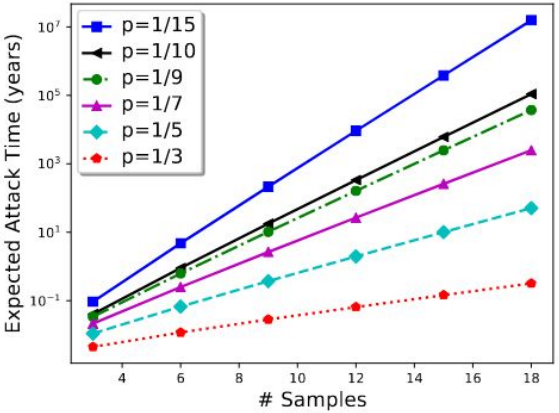
1. Must preserve the accuracy and precision of today's NTP. ✓
2. Must respect load distribution over timeservers. ✓
3. Must enhance NTP's security against malicious timeservers. ✓

Results

The difference allowed between the primary and secondary process is minimal.

Tinkering with the allowed discrepency changes how long an attack takes.

The number of Ananke servers polled by the secondary process also increases the time for a successful attack to occur.



p represents the probability of an Ananke server being controlled by an attacker.

# Samples is number of Ananke servers sampled.

Criticism

Flaws in Ananke

Organisations hosting Ananke-approved timeservers are still vulnerable to malicious internal actors.

Ananke-approved timeservers with alternate vulnerabilities are at risk to manipulation by malicious actors.

Malicious actors with MITM capabilities are still able to delay/discard traffic to manipulate client time.

No encryption allows malicious actors with MITM capabilities to modify traffic.

No authentication is provided, BGP IP-prefix hijacking and DNS cache poisoning are still possible.

Concluding Remarks

Ananke may not be an adequate solution to timeshift attacks in NTP.

Other standards such as Network Time Security may be more complete.

Lack of comparison with initial findings by the authors.

Still provides excellent insight into timeshift attacks within NTP.

Questions?

A link to the original paper:
https://www.ndss-symposium.org/wp-content/uploads/ndss2021_1A-2_24302_paper.pdf

Further information:
https://www.eecis.udel.edu/~mills/exec.html
https://datatracker.ietf.org/doc/html/rfc5905

Images:
https://www.pbtech.co.nz/fileslib/_20171206105759_Barebones.png
https://i.pinimg.com/originals/6f/8f/64/6f8f64e339935653a8d722d5fdd84401.jpg
https://media.istockphoto.com/photos/side-view-of-smartphone-with-blank-screen-and-modern-frame-less-on-picture-id1144491623?k=20&m=1144491623&s=612x612&w=0&h=VY5LE6QFykYcGWRgNOISYaLTxN4OzsBxpj-pJQLh1I0=
https://www.atomic-clock.galleon.eu.com/images/rackmountservergps.gif