Serwerem czasu nazywamy serwer komputerowy, pobierający czas z zewnętrznych źródeł i dystrybuuje go do innych urządzeń w sieci. Udostępniają bardzo precyzyjne dane czasowe, dokładność zależy od źródła czasu, z którego serwer korzysta. Serwer czasu może być używany jako lokalny lub internetowy.

Serwery wykorzystują rożne źródła zewnętrzne do synchronizacji czasu, takie jak:

- zegary atomowe,
- odbioniki czasu GNSS (Global Navigation Satellite System),
- oscylatory rubinowe,
- oscylatory cezowe.
- zegary wodorowe

Są to zegary o bardzo dużej precyzji, rzędu nanosekund, co pozwala na synchronizację czasu w sieciach komputerowych, telekomunikacyjnych, itp.

0.1 Protokoły synchronizacji czasu

Serwery te Wykorzystują różne protokoły sieciowe do synchronizacji czasu, takie jak:

- NTP (Network Time Protocol) - Wysyła okresowo pakiety synchronizacji czasu do serwerów w sieci i odpowiednim dostosowywaniu zegarów lokalnych. Jest to najpopularniejszy protokół synchronizacji czasu w sieciach komputerowych, jest on wspierany przez większość systemów operacyjnych.
- PTP (Precision Time Protocol) - jest bardziej precyzyjną alternatywą NTP i jest używany w systemach o wysokiej precyzji. Najczęściej stosowany w sieciach przemysłowych oraz przy badaniach naukowych. Jest w stanie osiągnąć dokładność synchronizacji zegarów do poniżej mikrosekundy.
- Algorytm Berkeley - to algorytm synchronizacji czasu opracowany na Uniwersytecie Kalifornijskim w Berkeley. Jego działanie polega na pomiarze szybkości dryfowania zegara między serwerami, często jest łączony z protokołem NTP.
- GPS - wykorzystuje odbiorniki GPS do synchronizacji zegarów na różnych serwerach. Zapewnia bardzo dokładne sygnały czasu. Czas ten można wykorzystać do synchronizacji zegarów serwerów podłączonych do tego samego odbiornika GPS.

Każdy z tych protokołów ma swoje nastepujace wady i zalety:

- NTP - Główną zaletą jest niezawodność i dokładność, co sprawia, że nadaje się do szerokiego zakresu zastosowań. Jednak NTP nie jest tak dokładny jak PTP i może synchronizować zegary z dokładnością do kilku milisekund. W związku z tym, że jest to leciwy protokół, nie jest najbardziej bezpiecznym rozwiązaniem, może być podatny na niektóre rodzaje ataków, takie jak ataki typu man-in-the-middle. Protokół istnieje już bardzo długo, więc dobrze znany i jest bardzo łatwy do obsługi.
- PTP - porównując do NTP, PTP jest bardziej precyzyjny i może synchronizować zegary z dokładnością do kilku mikrosekund. Jednak ma zdecydowanie większe wymagania sprzętowe(specjalistyczny sprzęt) i konfiguracyjne, co sprawia, że jest bardziej skomplikowany w użyciu.
- Algorytm Berkeley - możne być używać w połączeniu z NTP. Jedną z głównych zalet tego algorytmu jest to, że może synchronizować zegary z dokładnością do kilku mikrosekund, dzięki czemu nadaje się do wielu zastosowań. Podobnie jak w PTP wymaga on specjalistycznego sprzętu, co sprawia, że jest bardziej skomplikowany w użyciu i droższy.
- GPS - najbardziej precyzyjny z wymienionych protokołów, może synchronizować zegary z dokładnością do kilku nanosekund. Jest jednak nie zalecany do zastosowań wewnątrz pomieszczeń, ze względu na konieczność widoczności satelitów GPS i wymaga odbiornika GPS.

Z wyżej wymienionych protokołów, NTP jest najczęściej stosowany w sieciach komputerowych, dlatego też wydaje się być najlepszym wyborem do synchronizacji zegara nixie. Alternatywnym rozwiązaniem mozę być wykorzystanie własnego serwera który by zwracał czas wykorzystując REST API, ale wymaga to posiadania własnego serwera i jest zależne od jego działania.

0.2 Struktura serwerów w protokole NTP

Synchronizacji NTP wykorzystuje uporządkowaną strukture gałęziowa STRATUM. Zasada hierarchii wygląda nastepujaco: urządzenia warstwy STRATUM N mogą być serwerami czasu dla warstwy STRATUM N+1, ale nie na odwrót. Komputery STRATUM N mogą być również klientami urządzeń warstwy STRATUM N-1 itd.

Struktura ta ma na celu uporządkowanie i wprowadzenie hierarchii priorytetów urządzeń, zgodnie z ich rzeczywistym przeznaczeniem i funkcją. Aby nie nadmiernego skomplikowania systemu i związanych z tym opóźnień, ilość warstw została ograniczona do 16(STRATUM 0 - STRATUM 15).

Niektóre warstwy mają specjalne właściwości. Warstwa STRATUM 0 służy wyłącznie dla wzorców czasu, czyli zegarów atomowych, satelitarnych, itp. będących faktycznym źródłem czasu. Połączenie ze źródłem nie jest sieciowe, a zazwyczaj odbywa się za pomocą specjalnych interfejsów sprzętowych.

STRATUM 1 oraz STRATUM 2 stanowią najwyższe warstwy NTP i powinny być wykorzystane w przypadku dużych serwerów wysokiej jakości, superkomputerów lub sprzętowych serwerów czasu. Pozostałe warstwy są przeznaczone dla urządzeń lokalnych, takich jak komputery, routery, itp.

Numer STRATUM mówi jak daleko od wzorca czasu znajduje się dany serwer. Im niższy numer, tym bliżej źródła czasu. W rozbudowanych sieciach poziom STRATUM nie ma znaczącego wpływu na jakość synchronizacji i precyzję uzyskiwanego czasu.

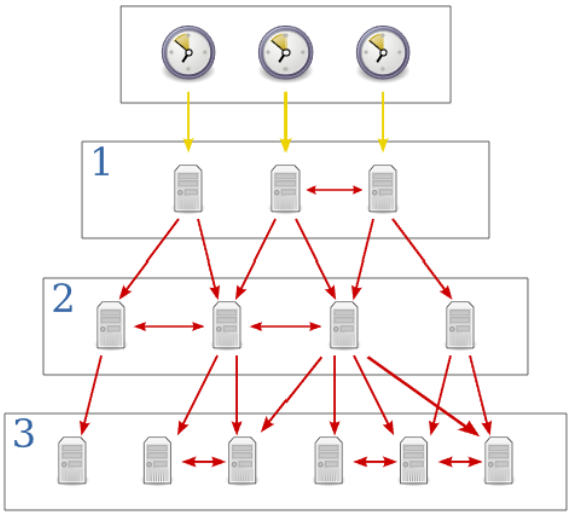

Figure 0.1: Struktura serwerów czasu w protokole NTP

W przypadku zegara nixie poziom STRATUM nie ma większego znaczenia, ponieważ zegar nie wymaga bardzo precyzyjnego czasu, chociaż oczywiście zależy, jak precyzyjny czas będzie wyświetlany, ale w przypadku zegara na 6 cyfrach, różnica w czasie rzędu kilku milisekund nie będzie zauważalna.

0.3 Zasada działania protokołu NTP

NTP różni się od typowego protokołu komunikacyjnego. Nie transmituje on bowiem absolutnej wartości czasu, lecz przekazuje informacje o opoznieniach i korelacjach czasowych w regularnych odstępach czasu, jakie zachodzą w sieci TCP/IP. Protokół wyróżnia się dopiero przy stosowaniu wielu źródeł czasu jednocześnie, wykorzystuje od wtedy algorytm analizy statystycznej czasu oparty na metodzie DTS (Dynamic Time Scales).

NTP wykorzystuje pakiety UDP o długości 72 bajtów na porcie 123, które są okresowo wymieniane co 2^{τ} sekund, gdzie τ wynosi od 4 (16s) do 17 (36h). Pozwala to klientom serwera, wyliczać opóźnienie względem idealnego czasu UTC. Znając aktualne opóźnienie w odniesieniu do czasu UTC, klient NTP sam kalibruje swój zegar lokalny, która polega na płynnym przyspieszaniu lub spowalnianiu pracy lokalnego zegara programowego. Przy różnicach czasu przekraczających 128ms, stosowana jest metoda step, która polega na skokowym przesunięciu zegara o określoną wartość. Dzieki temu każdy z klientów, asymptotycznie zmierza do czasu pochodzącego z wzorcowego zegara czasu UTC.

Sam pakiet NTP opisany jest w następujący sposób:

LI	VN	Mode	Stratum	Poll	Precision
Root Delay					
Root Dispersion					
Reference Identifier					
Reference Timestamp					
Originate Timestamp					
Receive Timestamp					
Transmit Timestamp					
Authenticator					

Table 1: NTP – format komunikatu

- LI – wskaźnik sekund przestępnych
- VN – (Version Number) numer wersji protokołu
- Mode – tryb pracy
- Stratum – warstwa, w której funkcjonuje komputer będący nadawcą komunikatu
- Poll interval – okres pomiędzy kolejnymi aktualizacjami czasu
- Precision – określenie dokładności zegara komputera wysyłającego dany komunikat
- Root Delay – opóźnienie pomiędzy nadawcą a serwerem warstwy 1
- Root Dispersion – maksymalny błąd pomiędzy zegarem lokalnym a serwera warstwy 1
- Reference Identifier – identyfikator źródła czasu, względem którego następuje synchronizacja
- Reference Timestamp – pole zawierające pomocnicze informacje o czasie poprzedniej synchronizacji
- Originate Timestamp – pole zawierające czas wysłania żądania przez klienta
- Receive Timestamp – czas odebrania komunikatu od klienta
- Transmit Timestamp – czas wysłania odpowiedzi do klienta
- Authenticator – informacje uwierzytelniające zarówno klienta, jak i serwer czasu

- Root Dispersion – maksymalny błąd pomiędzy zegarem lokalnym a serwera warstwy 1
- Reference Identifier – identyfikator źródła czasu, względem którego następuje synchronizacja
- Reference Timestamp – pole zawierające pomocnicze informacje o czasie poprzedniej synchronizacji
- Originate Timestamp – pole zawierające czas wysłania żądania przez klienta
- Receive Timestamp – czas odebrania komunikatu od klienta
- Transmit Timestamp – czas wysłania odpowiedzi do klienta
- Authenticator – informacje uwierzytelniające zarówno klienta, jak i serwer czasu