Nasjonal sikkerhetsmyndighet

IT-veiledning for ugraderte systemer nr. 9 (U-09).
Oppdatert: 2016-02-25

Autoritativ synkronisering av tid

Sikring av Network Time Protocol

Hvordan oppnå tillit til at datamaskiner har synkronisert og korrekt tid fra en autoritativ kilde.

Dette dokumentet beskriver hvordan Network Time Protocol (NTP) virker og hvordan man kan sikre denne protokollen for å oppnå høyere tillit til at tids-synkroniseringen er korrekt og kommer fra en autoritativ kilde. Målgruppen for denne veiledningen er personer som administrerer IT-systemer hvor tillit til synkronisert og korrekt tid er viktig.

Nasjonal sikkerhetsmyndighet

Nasjonal sikkerhetsmyndighet er tverrsektoriell fag- og tilsynsmyndighet innenfor forebyggende sikkerhetstjeneste i Norge og forvalter lov om forebyggende sikkerhet av 20. mars 1998. Hensikten med forebyggende sikkerhet er å motvirke trusler mot rikets selvstendighet og sikkerhet og andre vitale nasjonale sikkerhetsinteresser, primært spionasje, sabotasje og terrorhandlinger. Forebyggende sikkerhetstiltak skal ikke være mer inngripende enn strengt nødvendig, og skal bidra til et robust og sikkert samfunn.

Hensikt med veiledning

NSM sin veiledningsvirksomhet skal bygge kompetanse og øke sikkerhetsnivået i virksomhetene, gjennom økt motivasjon, evne og vilje til å gjennomføre sikkerhetstiltak. NSM gir jevnlig ut veiledninger til hjelp for implementering av de krav sikkerhetsloven stiller. NSM publiserer også veiledninger innen andre fagområder relatert til forebyggende sikkerhetsarbeid.

Postadresse
Postboks 814
1306 Sandvika

Sivil telefon/telefax
+47 67 86 40 00/+47 67 86 40 09

Militær telefon/telefaks
515 40 00/515 40 09

Webadresse
nsm.stat.no
E-postadresse
post@nsm.stat.no

Innhold

1 Innledning

Network Time Protocol (NTP) er en nettverksprotokoll for synkronisering av klokker i datamaskiner. Protokollen har blitt utviklet siden 1985, og den nåværende versjonen, NTPv4 [1], er i bruk på mange plattformer, som for eksempel Windows, GNU/Linux, OS X og integrerte enheter (*appliances*).

NTP har innebygde algoritmer for å etterstrebe synkronisert tid, men mangler gode sikkerhetsfunksjoner for beskyttelse mot manipulering. Tidsmanipulasjon kan ha alvorlige konsekvenser, som for eksempel innloggingsproblemer i Windows-domener, utdaterte nettleser-sertifikater som vil bli aksepterte som gyldige, *HTTP Strict Transport Security* (HSTS)-flagg som ignoreres, og tjenestebrudd i *Supervisory Control And Data Acquisition* (SCADA)-infrastruktur. I tillegg har NTP vært hovedkomponenten i flere store angrep, såkalte *reflected amplification attacks*, hvor NTP-servere har blitt manipulert til å bidra i tjenestenektangrep [2].

Formålet med denne veiledningen er å gi anbefalinger om hvordan man kan etablere sikker tidssynkronisering. Ved å implementere anbefalingene vil risikoen for at uvedkommende kan manipulere tiden på datamaskiner reduseres, mens tilliten til at tiden er korrekt og kommer fra en autoritativ kilde vil øke. Dokumentet er derimot ikke en konfigurasjons- eller driftsveiledning for NTP.

Autoritativ synkronisering av tid er ønskelig i innloggingssystemer som Kerberos, i distribuerte databasesystemer, når man gjør hendelses-korrelering ved hjelp av logger og/eller tidsstempler i filer, i systemer hvor *Domain Name System Security Extensions* (DNSSEC) benyttes, i systemer hvor *Internet Protocol Security* (IPSec) eller *Public Key Infrastucture* (PKI) benyttes, og i SCADA-systemer. Foretak som tilbyr investeringstjenester er fra 2017 pålagt av EU-direktiv 2014/65/EU (MiFID II/MiFIR) [3] å ha synkronisert tid for rapporteringsverdige hendelser.

2 Oversikt

Denne veiledningen har fire kapitler, samt vedlegg.

Kapittel 3 beskriver hva NTP er, hvordan NTP virker, samt alternative tidssynkroniseringsprotokoller.

Kapittel 4 beskriver tiltak for å sikre et IT-system som benytter NTP til tidssynkronisering.

Vedlegg A inneholder en punktvis oppsummering av de anbefalte sikringstiltakene for NTP.

Vedlegg B beskriver to eksempel-arkitekturer med NTP-servere, en arkitektur som illustrerer «best practice» og en arkitektur som er mindre kompleks, og som derfor har større sikkerhetsrisiko.

Vedlegg C inneholder en eksempel-konfigurasjon for NTP-servere som benytter Microsoft Windows.

Vedlegg D inneholder en eksempel-konfigurasjon for NTP-servere som benytter GNU/Linux.

Vedlegg E inneholder en tabell over NTP-versjonene og deres spesifikasjoner (*RFC-er*).

Vedlegg F inneholder en tabell over utvalgte tidskilder og noen av deres forskjeller.

Vedlegg G inneholder dette dokumentets eksterne referanser.

Vedlegg H inneholder dette dokumentets historikk.

Kontaktpunkt for denne veiledningen er si@nsm.stat.no. Kommentarer og innspill mottas med takk.

3 Om NTP

NTP benyttes for å oppnå synkroniserte klokker på flere datamaskiner, enten internt i en virksomhets nettverk, eller globalt, og dermed korrekt i samsvar med *Coordinated Universal Time* (UTC).

3.1 Arkitektur

NTP opererer typisk i en hierarkisk klient/server-arkitektur hvor en klient synkroniserer sin klokke mot en eller flere servere, men andre arkitekturer er også støttet. Nøyaktigheten til NTP avhenger av flere faktorer, men kan i lokalnett bli så nøyaktig som under ett millisekund. Nivåene i NTP-hierarkiet har forskjellige *stratum-nummer*.

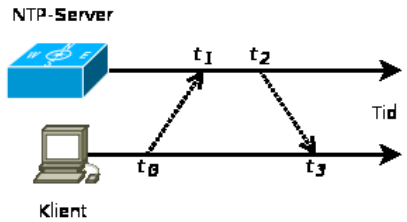
Stratum	Definisjon
0	Tidskilde, som for eksempel GPS-mottager, krystall-oscillator eller atom-oscillator.
1	Primær server, det vil si en server med direkte tilkobling til en tidskilde.
2 - 15	Sekundære servere, det vil si servere som synkroniserer sine klokker mot andre servere.
16	Usynkronisert server. Klienter vil ignorere Stratum 16-servere.
17 - 255	Reservert for eget bruk.

Tabell 1: Oversikt over Stratum-nivåer og betydninger.

Servere med direktekoblinger til tidskilder kalles *Stratum 1-servere*. Servere som ikke har egne tidskilder, men er synkroniserte med Stratum 1-servere kalles *Stratum 2-servere*, og så videre med stadig mindre nøyaktig tid, opp til Stratum 15. Stratum 0 brukes ikke om servere, men om tidskilder. Stratum 16 benyttes om ikke-synkroniserte servere, og Stratum 17 til 255 er reservert for eget bruk.

3.2 Protokollbeskrivelse

Synkronisering med NTP fungerer slik at klienten som skal stille sin klokke sender en UDP-pakke med tre forskjellige tidsstempler som skal fylles inn, t_0 til t_2, til serveren. Klienten setter t_0 til sin tid før pakken sendes til serveren. Serveren setter t_1 til sin tid ved mottak av pakken fra klienten, og t_2 til sin tid ved retur av pakken til klienten. Til slutt fastslår klienten tiden ved mottak av pakken fra serveren, t_3. Figur 1 viser hvordan NTP-pakken blir sendt frem og tilbake mellom klient og server, og når de forskjellige tidsstemplene settes.



Figur 1: Synkronisering av tid på klient mot NTP-server.

De fire tidsstemplene, t_0 til t_3, benyttes av klienten for å regne ut prosesseringstid på serveren, reisetid i nettverket og tidsforskjellen mellom klientens klokke og serverens klokke. Når klienten har gjort disse utregningene kan den justere sin klokke til å samsvare med serverens klokke. Prosessen med sending og mottak av tidsstempler gjentas flere ganger ved hver synkronisering, og ofte mot forskjel-

lige servere (*pooling*). På denne måten kan statistisk analyse bidra til å øke nøyaktigheten av synkroniseringen.

Uten sikring av NTP er det mulig for ondsinnede aktører å kompromittere:

- **Autentisiteten**: Opptre på vegne av tiltenkt NTP-server uten å bli oppdaget.
- **Integriteten**: Forsinke forespørslene med tidsstemplene slik at synkroniseringen blir unøyaktig eller endre forespørslene med tidsstemplene slik at synkroniseringen blir manipulert.
- **Tilgjengeligheten**: Stoppe forespørslene med tidsstemplene slik at synkroniseringen aldri finner sted eller overbelaste servere eller klienter med falsk trafikk slik at synkroniseringen aldri finner sted.

En utfyllende beskrivelse av sårbarheter finnes i RFC 7384 [4].

3.3 Alternative protokoller

NTP er ikke den eneste nettverksprotokollen for synkronisering av tid. Under følger en oversikt over noen av alternativene.

Simple Network Time Protocol (SNTP) er en forenklet versjon av NTP for datamaskiner som ikke trenger like høy presisjon som NTP tilbyr. Anbefalingene som gis i denne veiledningen vil kunne anvendes i systemer som benytter SNTP.

Precision Time Protocol (PTP) er en tidssynkroniseringsprotokoll med høy nøyaktighet på lokalnettverk som mål. PTP benytter *IP Multicast* over UDP og mangler tilstrekkelige sikkerhetsmekanismer for å kunne tilby autoritativ synkronisering av tid. PTP kan derfor ikke anbefales som et sikkert alternativ til NTP.

tlsdate [5] er en protokoll som synkroniserer klientens klokke via tidsstempler som utveksles i *Transport Layer Security* (TLS)-protokollen. *tlsdate* kan autentisere serveren, og har derfor bedre sikkerhet enn NTP, men anbefales likevel ikke. For det første er ikke TLS laget for synkronisering av tid, noe som gir lavere nøyaktighet enn NTP. For det andre avhenger *tlsdate* av tidsstempler som i neste versjon av TLS er på vei ut, noe som innebærer at *tlsdate* vil slutte å fungere i fremtiden.

Network Time Security (NTS) er en foreslått protokoll som kan gi beskyttelse av autentisitet og integritet ved synkronisering [6]. Dersom NTS blir en akseptert og utbredt standard med tilgjengelige implementasjoner anbefales det å benytte NTS.

Ingen av de tilgjengelige protokollene har fullverdige sikkerhetsmekanismer for å oppnå synkronisert og korrekt tid fra en autoritativ kilde. Det er derfor viktig å sikre NTP.

4 Anbefalte tiltak

En virksomhet bør vurdere om det er tilstrekkelig at virksomhetens datamaskiner har synkronisert tid eller om også korrekt tid er viktig. Synkronisert tid er viktig i alle systemer. Korrekt tid er viktig i systemer som interagerer med brukere og/eller andre eksterne systemer, men er mindre viktig i isolerte systemer uten kontakt med omverdenen. Autoritativ tid, det vil si tillit til at tiden kommer fra riktig kilde, er viktig i alle tilfeller. For å øke tilliten til synkronisert, korrekt og autoritativ tid er det flere tiltak som bør iverksettes.

4.1 Tilpass server- og nettverkskapasiteten

Fordi en overbelastet NTP-server vil gi unøyaktig tid til sine klienter er det viktig å estimere hvor mange klienter og klientforespørsler man kan forvente i et *worst-case-scenario*. Integrerte enheter (*appliances*) oppgir som regel sin maksimale kapasitet i dokumentasjonen og man kan referere til denne for å verifisere at belastningen man forventer kan håndteres. Dersom man ikke bruker integrerte enheter med tilgjengelig dokumentasjon er man selv avhengig av å gjøre målinger for å få bekreftet at belastningen man forventer kan håndteres.

Nettverksinfrastrukturen må også være i stand til å håndtere NTP-trafikken, noe som er særlig viktig dersom mye annen trafikk går over de samme linkene eller avstandene mellom klienter og servere er store. Høy latenstid (*latency*) vil gi utslag i form av unøyaktig tidssynkronisering, mens kortvarige variasjoner (*jitter*) håndteres av NTP-protokollen ved hjelp av statistisk analyse. NTP-pakkene er små, typisk 100 bytes, og krever derfor ikke mye båndbredde.

4.2 Velg en nøyaktig tidskilde

NTP sørger for at klokken på klienten er synkronisert med klokken på serveren, men kan ikke sørge for at disse to klokkene er korrekte. For å få korrekt tid er det nødvendig med en ekstern tidskilde på serveren. Det finnes flere mulige tidskilder. Når man skal sette opp en NTP-server er det derfor viktig å velge en kilde som oppfyller virksomhetens krav til nøyaktighet.

Typiske tidskilder er Cesium-, Rubidium-, og krystallbaserte oscillatorer, samt GPS-mottagere. Tidskildene har forskjellige fordeler og ulemper når det kommer til nøyaktighet, pris, størrelse, varmeutvikling og strømbehov. Oscillatorer, en mye brukt klasse av tidskilder, kan ikke stille en NTP-servers klokke initialt, men kun indikere en puls av tiden som går.

Cesium anbefales som primær tidskilde for alle virksomheter hvor nøyaktig tid er essensielt. En Cesium-oscillator er den mest nøyaktige tidskilden fordi Cesium-atomet danner grunnlaget for definisjonen av ett sekund. 1 sekund = 1 192 631 770 svingninger i et Cesium-133-atom. Cesium-oscillatorer er også i liten grad påvirket av eksterne faktorer som temperatur og luftfuktighet. Ulempen med Cesium-oscillatorer er at de er kostbare og krever mer plass enn andre tidskilder.

GPS-mottagere anbefales ikke som primær tidskilde uten grundig risikovurdering. GPS-mottagere kan likevel benyttes for å stille NTP-serveres klokke initialt. GPS-mottagere er nøyaktige tidskilder ettersom de mottar signaler fra GPS-satellittene som alle har Cesium-oscillatorer ombord. De mottar også, som oftest, signaler fra flere GPS-satellitter på samme tid, noe som øker nøyaktigheten. Videre er signalene som mottas ikke bare pulser, men fullstendige tidsstempler. Dette gjør at GPS-mottagere både kan stille klokken initialt, samt holde nøyaktig tid etter dette. Dette er unikt for GPS-mottagere sammenlignet med de andre tidskildene nevnt i denne veiledningen. GPS-mottagere er

også rimelige, sammenlignet med atom-baserte oscillatorer. Ulempen er at de er enkle å manipulere og å forstyrre (*jamme*).

Rubidium-oscillatorer anbefales kun som primær tidskilde dersom en Cesium-oscillator er utelukket. Dette er fordi Rubidium-oscillatorer er mindre stabile enn Cesium-oscillatorer.

Krystall-oscillator anbefales kun som sekundær tidskilde for å bidra til stabilitet ved korte bortfall av primær-kilden. Krystall-oscillatorer er den minst stabile av tidskildene nevnt over, og kan i enkelte tilfeller avvike med flere millisekunder per dag. Dette er fordi krystall-oscillatorer er påvirket av ytre faktorer som for eksempel temperatur.

Et kompromiss ved valg av tidskilde kan være å benytte en krystall-oscillator som over tid vil ha store avvik fra UTC, men kompensere for dette ved å synkronisere NTP-serveren med en GPS-mottager med jevne mellomrom, for eksempel månedlig. Hyppigheten på synkroniseringen med GPS-mottager må i så fall avgjøres av virksomhetens toleranse for unøyaktighet i forhold til UTC og krystall-oscillatorens *drift*. Forventet drift oppgis som regel i integrerte enheters (*appliances*) dokumentasjon.

Dersom man ikke bruker integrerte enheter med tilgjengelig dokumentasjon kan man selv finne *driften* hver gang man synkroniserer med GPS-mottageren og justere synkroniseringsintervallet deretter. Selv om virksomhetens datamaskiner i dette kompromisset ikke vil være i samsvar med UTC til enhver tid vil de være synkroniserte seg imellom.

4.3 Opprettholde tilstrekkelig redundans

Det er anbefalt å ha minst tre Stratum 1-servere som synkroniserer mot hverandre (*peering*). Først og fremst for å finne ut om utstyr feiler, og hvilket utstyr, men også ved bruk av GPS-mottager som tidskilde ettersom GPS-signaler enkelt kan manipuleres eller forstyrres (*jammes*). Ved bruk av færre enn tre redundante servere vil det ikke være mulig å avdekke hvilket utstyr som eventuelt feiler. I følgende tenkte scenario sier Server A at klokken er 13:37 og Server B at klokken er 13:38. Man kan vite at en av serverne tar feil, men ikke hvilken. Det er nødvendig med en Server C, som kan bidra til å avklare om det er Server A eller B som feiler.

Stratum 1-servere bør konfigureres slik at de synkroniserer mot hverandre like ofte som de synkroniserer mot tidskildene sine. Dette gjør at avvikende tidskilder filtreres bort.

Det er anbefalt å ha geografisk separasjon mellom serverne. Dette for å redusere konsekvensene av eventuelle brudd i infrastruktur, som for eksempel strømnett eller datanettverk. Det er også viktig med geografisk separasjon for å gjøre det vanskeligere å manipulere eller forstyrre (*jamme*) GPS-mottagere i systemer hvor det er brukt. Det anses som lite sannsynlig at flere, geografisk separerte GPS-mottagere *jammes* tilfeldig på samme tid. I så fall er man utsatt for et målrettet angrep.

Redundans øker altså synkroniseringsnøyaktigheten og tilgjengeligheten til NTP, men det må en risikovurdering til for å bestemme ønsket mengde av redundans og eventuell geografisk separasjon.

4.4 Introduser et Stratum 2-nivå

Det anbefales å opprette Stratum 2-servere for å separere høytillits Stratum 1-servere fra klientmaskiner. En sterk separasjon kan øke sikkerheten ved å redusere angrepsflaten til de kritiske Stratum 1-serverne. En annen grunn til å opprette Stratum 2-servere er for å øke nøyaktigheten av synkroniseringen i lokale partisjoner av et IT-system. I et IT-system som er spredt på forskjellige lokasjoner kan det for eksempel være relevant med lokale Stratum 2-servere på hver lokasjon.

Tjenesteleverandører kan opprette kontraktsfestet, symmetrisk synkronisering (*peering*) mot andre pålitelige tjenesteleverandører. Slik synkronisering bør i så fall komme i tillegg til egne Stratum 1-servere. Det har ingen verdi for Stratum 2-servere å synkronisere mot ikke-tiltrodde NTP-servere.

Det er sjeldent ønskelig å opprette Stratum 3 - 15-servere da det gir ekstra kompleksitet uten ekstra verdi, og fordi hvert ekstra ledd mellom tidskilde og klient reduserer nøyaktigheten av synkroniseringen.

Det kan være aktuelt å opprette enveis-forbindelser, for eksempel med *IRIG-pulser* [7], for å unngå at de mest kritiske Stratum 1-serverne har datanettverksforbindelse med Stratum 2-serverne. Fordelen av økt separasjon må i så fall veies opp mot den ekstra kompleksiteten som innføres.

4.5 Sikre kommunikasjonen mellom klienter og servere

NTP kommuniserer uten autentisering og over UDP, noe som gjør det trivielt å endre avsenderadresse. Dette har vært en måte å utnytte NTP-servere til å gjøre tjenestenektangrep. NTP-forespørsler har blitt sendt til mange NTP-servere med falsk avsenderadresse som peker på et offer, noe som har resultert i overbelastning hos offeret som har måttet bruke betydelig båndbredde på svarene fra NTP-serverne.

Den ikke-autentiserte kommunikasjonen gjør det også mulig å gjøre såkalte *man-in-the-middle-angrep*, hvor pakker fanges opp på veien mellom klient og server og manipuleres. Det finnes også eksempler på angrep hvor ondsinnede aktører ikke behøver å være *man-in-the-middle* for å gjøre tidsmanipulasjons [8].

Autentisitet- og integritetsbeskyttelse av NTP-trafikk er særlig viktig hvis trafikken sendes over åpne datanettverk, som for eksempel Internet.

I et forsøk på å sikre kommunikasjonen mellom klient og server har NTP-protokollen støtte for to sikkerhetsmekanismer: *Symmetriske nøkler* (statiske nøkler) og *AutoKey* [9] (dynamiske nøkler). Sikkerhetsmekanismene er ment å gi autentisering og integritetsbeskyttelse, men ingen av disse mekanismene gir per i dag høy sikkerhet.

Symmetriske nøkler anbefales ikke ettersom meldingskodene som genereres på bakgrunn av nøklene er for svake. MD5 er NTPs standard algoritme for å generere slike meldingskoder, men enkelte implementasjoner støtter også SHA-1-algoritmen. Bruk av meldingskoder basert på disse algoritmene innebærer at ondsinnede aktører kan manipulere NTP-trafikken mens den fremstår som autentisert. I tillegg til svak sikkerhet er det betydelig manuell forvaltning forbundet med bruk av statiske nøkler ettersom disse må konfigureres manuelt på hver enkelt server og klient.

Dynamiske nøkler anbefales ikke ettersom AutoKey har flere kjente svakheter og kan enkelt kompromitteres [10]. Den største svakheten ved AutoKey er at «*server seedet*», som skal være hemmelig, kun består av 32 bit, og derfor er trivielt å finne med *brute force*. Dette gjør at ondsinnede aktører kan utgi seg for å være en autentisert, tiltrodd server.

Standardiserte, krypterte *Virtual Private Network* (VPN)-tunneler eller IPSec anbefales ettersom de kan gi god autentisitet- og integritetsbeskyttelse for NTP-trafikk. Med kryptert VPN eller IPSec vil man samtidig oppnå konfidensialitetsbeskyttelse, men dette anses som lite viktig for NTP-trafikk. Det er ikke trivielt å konfigurere slike tunneler for en tjeneste som NTP, men vil være et svært effektivt sikkerhetstiltak.

Kommunikasjonen til og fra NTP-servere bør sikres med brannmur som kun åpner for ønsket trafikk, det vil si UDP-pakker til og fra port 123. Dersom det lar seg gjøre bør man også begrense hvilke maskiner som får kommunisere med hverandre. Se for øvrig **Vedlegg B** for forslag til systemarkitekturer.

4.6 Gjennomfør herding og regelmessig vedlikehold av serverne

Det er viktig NTP-serverene kommer fra en seriøs leverandør med høyt sikkerhetsfokus. Dette innebærer blant annet at feil utbedres fortløpende, at leverandøren kan levere støtte i flere år og at det er sertifisert eller evaluert av tredjepart.

NTP-servere må konfigureres med tanke på sikkerhet (*herdes*) for å inngå i en sikker plattform. Det oppdages jevnlig nye sårbarheter i både operativsystem, NTP-protokollen og forskjellige NTP-implementasjoner [11], så alle NTP-servere må patches regelmessig. Videre bør angrepsflaten reduseres ved å slå av alle ubenyttede funksjoner og protokoller.

For å opprettholde sikker drift og vedlikehold må NTP-servere overvåkes. Logging og varsling må være påslått for å kunne følge opp sikkerhetsrelevante hendelser. Videre bør loggene sendes til et sentralisert system, som for eksempel ELK (*Elasticsearch, Logstash og Kibana*) [12] eller Splunk [13]. NSM er i ferd med å utgi en veiledning om hvordan man kan bruke ELK til loggkorrelering. Veiledningen vil publiseres på NSM sin webside.

Fjernadministrasjon av servere bør skje i en dedikert nettverkssone som kun er tilgjengelig for administratorer. Selv store og kjente aktører kan ha, og har hatt, alvorlige sårbarheter knyttet til sine administrasjonsgrensesnitt [14]. Fjernadministrasjon bør skje over sikre kanaler som HTTPS eller SSHv2.

4.7 Konfigurer klientene for optimalisert tjenesteoperasjon

Klienter bør sende flere forespørsler, for eksempel 8, ved hver synkronisering for å øke nøyaktigheten på synkroniseringen. Mange forespørsler gir NTP bedre utgangspunkt for sine beregninger.

Videre bør klientene sende forespørsler til flere servere (*pooling*) for å forbedre nøyaktigheten på synkroniseringen. Dersom flere servere involveres er det høyere sannsynlighet for at minst en er tilgjengelig, og bruk av forskjellige servere gir NTP bedre utgangspunkt for sine beregninger.

Klientene bør også konfigureres til å kun tillate små justeringer fra maskinens klokke ved hver synkronisering hvis det er mulig i den gitte implementasjonen. Dette reduserer sjansen for store, uriktige forskyvinger.

Intervallet mellom synkroniseringer bør avgjøres av klienten selv, basert på *drift*. Dette er standard oppførsel i NTP.

4.8 Verifiser at anbefalingene virker som tiltenkt

Det er viktig å påse at tiltakene er korrekt implementert og operative. For å verifisere dette kan man gjøre følgende tester:

- Bekreft i administrasjonsgrensesnittet til serverne at riktig tidskilde er aktiv og fungerende, for eksempel at GPS-mottageren kommuniserer med flere GPS-satellitter.
- Bekreft i administrasjonsgrensesnittet til serverne at eventuelle *peering*-assosiasjoner er aktive.

- Bekreft at NTP-serverne er utilgjengelige for uautoriserte klienter, hvis sikre kanaler kreves.
- Bekreft i administrasjonsgrensesnittet til serverne at nyeste versjon benyttes.
- Bekreft i administrasjonsgrensesnittet at alle ubenyttede tjenester er avslått.
- Bekreft i administrasjonsgrensesnittet at fjernadministrasjon bare er mulig i dedikert nettverkssone og over sikre forbindelser.
- Bekreft at klientene sender det korrekte antall forespørsler og at forespørslene når alle serverne, for eksempel ved hjelp av verktøyene *ntpq -p*, *tcpdump* eller *Wireshark*.

Dersom overnevnte punkter er verifisert kan man med rimelig sikkerhet si at anbefalingene virker som tiltenkt.

Vedlegg A Oppsummering

Vi har sett at synkronisert tid er viktig og at det er nødvendig å sikre synkroniseringen for å unngå alvorlige konsekvenser som tjenestebrudd. Det er også viktig å sikre synkroniseringen for å oppnå at tiden som settes på datamaskinene kommer fra en autoritativ kilde. Anbefalingene gitt i denne veiledningen kan oppsummeres på følgende måte:

- Tilpass server- og nettverkskapasiteten til et *worst-case-scenario*.
- Velg en nøyaktig tidskilde som Cesium-oscillator.
- Opprettholde tilstrekkelig redundans med flere servere.
- Introduser et Stratum 2-nivå.
- Sikre kommunikasjonen mellom klienter og servere med VPN, IPSec eller tilsvarende.
- Gjennomfør herding og regelmessig vedlikehold av serverne.
- Inkluder NTP-serverne i virksomhetens systemovervåkningsverktøy hvor sikkerhetsrelevante hendelser følges opp.
- Konfigurer klientene for optimalisert tjenesteoperasjon ved å øke antall forespørsler ved hver synkronisering, samt ved å sende forespørslene til flere servere.
- Verifiser at anbefalingene virker som tiltenkt.

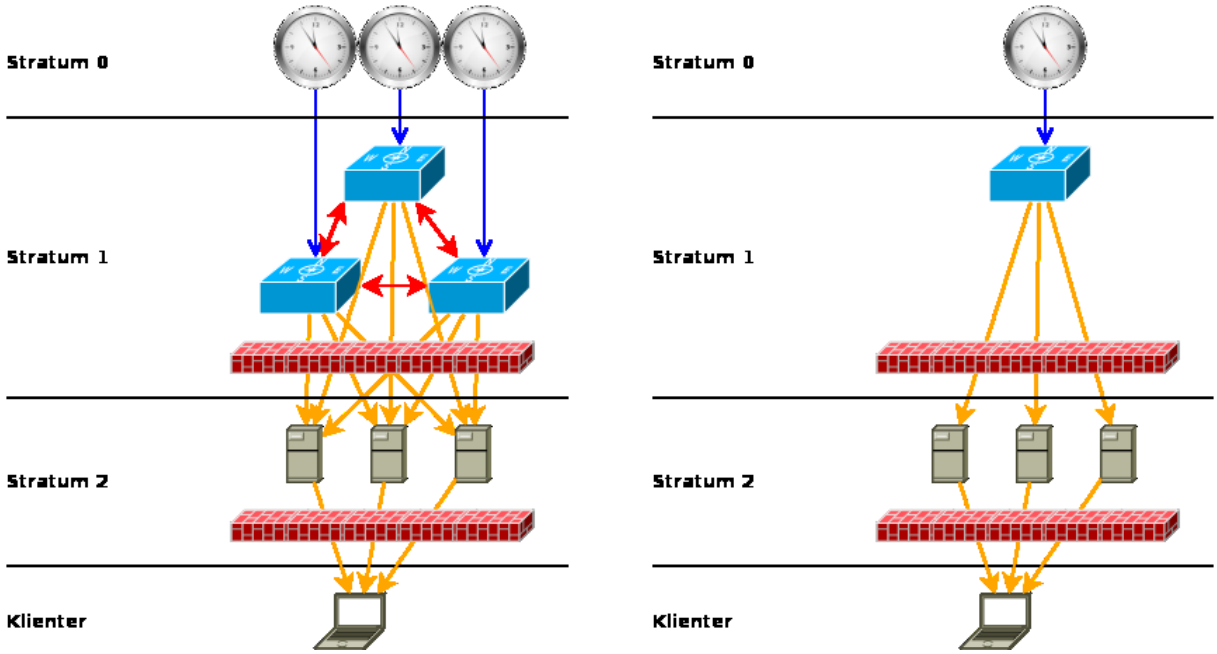
To særlig viktige anbefalinger er å sikre kommunikasjonen mellom klienter og servere, samt gjennomføre herding og regelmessig vedlikehold av serverne.

Vedlegg B Eksempel-arkitekturer

Figur 2 illustrerer to eksempelarkitekturer med NTP-servere.

Eksempelet til venstre skal illustrere «*best practice*» for virksomheter med høye sikkerhetskrav. Denne arkitekturen har redundans og brannmur-filtrering i alle ledd.

Eksempelet til høyre er mindre komplekst og har derfor større risiko for feil og utilgjengelighet. Hvis GPS-mottager benyttes som tidskilde i en slik arkitektur er det ikke anbefalt å ha denne tilkoblet til enhver tid, men resynkronisere Stratum 1-serveren til GPS-mottageren ved faste intervall, slik det er beskrevet i kapittel 4.2.



Figur 2: To eksempler på NTP-arkitekturer.

Blå linjer indikerer direkte-tilkoblinger ved hjelp av for eksempel seriell-port. Rød linjer indikerer *peering*-tilkoblinger. Oransje linjer indikerer klient/server-tilkoblinger.

Brannmur-regelen for å kun tillate NTP-trafikk er å blokkere alt annet enn UDP-pakker til og fra port 123, men hvis Stratum 2-serverne har andre funksjoner enn NTP, som for eksempel domenekontrollere, må det også åpnes for den forventede kommunikasjonen for disse tjenestene. Dersom det lar seg gjøre anbefales det også å begrense hvilke maskiner som får kommunisere med hverandre.

Vedlegg C Eksempel-konfigurasjon (Windows)

Alle versjoner av Windows har siden Windows 2000 hatt «*Windows Time Service*», også kjent som «*W32Time»*, installert, og alle maskiner som er med i et *Active Directory-domene* er satt opp med NTP som standard. I slike domener er domenekontrollerne (serverne) automatisk konfigurert til å synkronisere eksternt mot Microsoft sin NTP-server og tilby sin egen NTP-server internt i domenet, mens domenemedlemmene (klientene) er automatisk konfigurert til å synkronisere mot domene-kontrollerne.

I eksempel-arkitekturene i Vedlegg B er domenekontrollerne typisk Stratum 2-servere som synkroniserer mot egne, dedikerte Stratum 1-servere. Konfigurasjonen under er derfor tenkt brukt på Windows-domenekontrollere som fungerer som Stratum 2-servere. Målet med konfigurasjon er å endre domenekontrollernes NTP-server, bort fra Microsoft sin server over til egne, dedikerte NTP-servere.

NTP-tjenesten i Windows kan konfigureres via registeret, Group Policy editor, eller kommandolinje-verktøyet *w32tm.exe*. I følgende eksempler benyttes sistnevnte.

```
# Synkroniser mot tre egne NTP-servere
w32tm.exe /config /manualpeerlist:"ntp0.VIRKSOMHET.no ntp1.VIRKSOMHET.no ntp2.VIRKSOMHET.no"
w32tm.exe /config /syncfromflags:MANUAL

# Får tidstjenesten til å laste ny konfigurasjon
w32tm.exe /config /update

# Viser status og konfigurasjon for tidstjenesten, egnet for verifikasjon
w32tm.exe /query /status /verbose
w32tm.exe /query /configuration /verbose

# Synkroniserer klokken nå
w32tm.exe /resync
```

Mer informasjon om w32tm.exe finnes på Microsoft TechNet [15].

Vedlegg D Eksempel-konfigurasjon (Linux)

NTP benyttes i mange integrerte enheter (*appliances*) og på datamaskiner som benytter GNU/Linux som operativsystem. Disse benytter typisk tjenesten *ntpd* for å synkronisere datamaskiners klokker. *ntpd* kan som regel konfigureres via et grafisk grensesnitt eller direkte i konfigurasjonsfilen *ntpd.conf*. I følgende eksempel benyttes sistnevnte.

I eksempel-arkitekturene i Vedlegg B kan *ntpd* typisk bli brukt av både Stratum 1-serverne som integrerte enheter, og av Stratum 2-serverne. Konfigurasjonen under er tenkt brukt på GNU/Linux-servere som fungerer som Stratum 2-servere, men konfigurasjonen av Stratum 1-servere vil bli tilsvarende hvor nøkkelordet *server* er byttet ut med *peer*. Målet med konfigurasjon er å sette opp synkronisering til egne, dedikerte NTP-servere.

```
# NTP-konfigurasjon, typisk lagret i /etc/ntpd.conf


# Ikke tillat administrasjonstrafikk

# nomodify = Ikke tillat at ntpd-konfigurasjonen endres over nettverket

# noquery = Ikke tillat at ntpds status gis over nettverket

# nopeer = Ikke tillat nye server-assosiasjoner

restrict default nomodify noquery nopeer


# Sett sti til fil som holder oversikt over klokkens avvik fra serverne

driftfile /var/lib/ntp/ntp.drift


# Sett sti til generell loggfil

logfile /var/log/ntp


# Definer tre egne Stratum 1-servere

server ntp0.VIRKSOMHET.no iburst

server ntp1.VIRKSOMHET.no iburst

server ntp2.VIRKSOMHET.no iburst
```

Man kan teste at konfigurasjonen virker med kommandoen *ntpq –p* etter at konfigurasjonen er lagret og NTP-tjenesten er startet på nytt. På grunn av *restrict*-linjen vil dette bare virke fra samme maskin. Dette kan endres til å også virke fra andre maskiner, men da anbefales det at dedikerte administrasjonsnettverk benyttes. Mer informasjon om *ntpd* finnes i den offisielle NTP-dokumentasjonen [16].

Vedlegg E NTP-versjoner

Vedlagt er en oversikt over NTP-versjonene og deres spesifikasjoner (*RFC-er*).

Version	År	Spesifikasjon	Beskrivelse
NTPv0	1985	RFC 958	Første spesifikasjon.
NTPv1	1988	RFC 1059	En mer fullstendig spesifikasjon.
NTPv2	1989	RFC 1119	Introduksjon av sikkerhetsmekanismer.
NTPv3	1992	RFC 1305	Introduksjon av feilkilde-analyse.
NTPv4	2010	RFC 5905	Nåværende spesifikasjon, med bakoverstøtte.

Tabell 2: Oversikt over NTP-versjoner.

NTP utvikles kontinuerlig og en versjon 5 vil trolig komme på sikt.

Vedlegg F Tidskilder

Under vises en oversikt over utvalgte tidskilder og noen av deres forskjeller.

Type	Nøyaktighet	Strømforbruk	Størrelse
Cesium	Helt nøyaktig	~ 30 Watt	Svært stor
Rubidium	Svært nøyaktig	~ 20 Watt	Stor
GPS-mottager	Svært nøyaktig	~ 4 Watt	Liten, men krever utendørs antenne
Ovnskontrollert kvarts (OCXO)	Ganske nøyaktig	~ 0.6 Watt	Middels
Mikroprosessor-kompensert kvarts (MCXO)	Ganske nøyaktig	~ 0.04 Watt	Liten
Temperatur-kompensert kvarts (TCXO)	Noe unøyaktig	~ 0.05 Watt	Liten

Tabell 3: Oversikt over typiske tidskilder og noen av deres forksjeller.

Tabellen over er sortert etter nøyaktighet.

Vedlegg G Referanser

[1] D. Mills, U. Delaware, J. Martin, J. Burbank, W. Kasch, "Network Time Protocol Version 4: Protocol and Algorithms Specification", 2010, https://tools.ietf.org/html/rfc5905

[2] Forskjellige forfattere, "NTP Amplification Attacks Using CVE-2013-5211", 2014, https://www.us-cert.gov/ncas/alerts/TA14-013A

[3] Forskjellige forfattere, "Directive 2014/65/EU of the European Parliament and of the Council", 2014, http://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32014L0065

[4] T. Mizrahi, "Security Requirements of Time Protocols in Packet Switched Networks", 2014, https://tools.ietf.org/html/rfc7384

[5] J. Appelbaum, C. Grothoff, E. Jones, W. Drewry, "tlsdate", 2014, https://github.com/ioerror/tlsdate

[6] D. Sibold, S. Roettger, K. Teichel, "Network Time Security", 2015, https://tools.ietf.org/html/draft-ietf-ntp-network-time-security-08

[7] Forskjellige forfattere, "IRIG Standard 200-04", 2015, http://www.wsmr.army.mil/RCCsite/Documents/200-04_IRIG_Serial_Time_Code_Formats/200-04_IRIG_Serial_Time_Code_Formats.pdf

[8] A. Malhotra, I. E. Cohen, E. Brakke, S. Goldberg, "Attacking the Network Time Protocol", 2015, https://eprint.iacr.org/2015/1020.pdf

[9] B. Haberman, D. Mills, U. Delaware, "Network Time Protocol Version 4: Autokey Specification", 2010, http://tools.ietf.org/html/rfc5906

[10] S. Röttger, "Analysis of the NTP Autokey Procedures", 2012, http://zero-entropy.de/autokey_analysis.pdf

[11] Forskjellige forfattere, Security Notice, 2016, http://support.ntp.org/bin/view/Main/SecurityNotice

[12] Forskjellige forfattere, "Elastic", 2016, https://www.elastic.co/

[13] Forskjellige forfattere, "Splunk Inc.", 2016, http://www.splunk.com/

[14] T. Brown, M. Emery, "How Many Bugs Can A Time Server Have?", 2014, https://labs.portcullis.co.uk/download/HMBCATSHEMFC.pdf

[15] Forskjellige forfattere, "Windows Time Service Tools and Settings", 2012, https://technet.microsoft.com/en-us/library/cc773263(v=ws.10).aspx

[16] Forskjellige forfattere, "The NTP Public Services Project", 2015, http://support.ntp.org/

Vedlegg H Dokumenthistorikk

2015-04-21 Dokumentet ble opprettet.

2016-02-01 Intern høring.

2016-02-29 Publisering på nett.