Security aspects of time synchronization infrastructure

http://www.security.nnov.ru/advisories/timesync.asp

Aspects de la sécurité d'une infrastructure de synchronisation horaire

Traduction personnelle française (avec accord de l'auteur) : Jérôme ATHIAS

Un grand nombre de services dans un réseau d'entreprise moderne nécessitent que le temps soit synchronisé au sein du réseau ou par rapport au temps universel et peuvent devenir inopérants si un problème dans la synchronisation est rencontré. Ce qui suit constitue juste quelques exemples de services et de nécessité du temps.

Pour la synchronisation au sein du réseau :

- Licences/clés de logiciels expirées : ~en semaines
- Synchronisation de répertoires : ~en minutes
- Services collaboratifs : ~en minutes
- Services de planification : ~en minutes
- Services d'authentification (par exemple Kerberos) : ~en minutes
- Journalisation (Logs) : ~en secondes
- Production : ~en millisecondes

Pour la synchronisation avec le temps universel :

- Expiration de certificats : ~en mois
- Licences/clés de logiciels expirées : ~en semaines
- Expiration de comptes : ~en jours
- Heures de permission d'accès : ~en minutes
- Services collaboratifs : ~en minutes
- Services de planification : ~en minutes
- Authentification pour les intervenants principaux externes (par exemple les sites de confiance par Kerberos) : ~en minutes
- Authentification avec les périphériques physiques (par exemples des eTokens) : ~en minutes
- Journalisation (Logs) : ~en secondes
- Production : ~en millisecondes

Le but de cet article n'est pas de montrer l'importance d'une infrastructure de synchronisation du temps, mais de montrer certains aspects importants de l'implémentation d'une infrastructure de synchronisation du temps et de traiter de certains problèmes liés dans la vie de tous les jours.

Si certains de ces services sont utilisés au sein d'un réseau d'entreprise et sont considérés comme des services critiques, alors la synchronisation du temps se doit également d'être considérée comme un service critique. Chaque incohérence dans la synchronisation du temps qui provoque une erreur, légèreté d'administration, sabotage ou attaque réseau peut conduire les services dépendants à échouer et entraîner une perte de productivité ou de sécurité et amener à des dépenses supplémentaires. Un exemple très courant est qu'un mauvais réglage de l'horloge d'une station de travail peut empêcher l'utilisateur de se connecter à sa station de travail et de réaliser ses tâches courantes.

Il est très courant que la synchronisation du temps ne soit pas prise en compte dans l'architecture d'un réseau ou que seuls les mécanismes implémentés par les vendeurs de logiciels soient utilisés sans examiner ces mécanismes du point de vue de la stabilité, de la sécurité et de la conformité avec l'attente de la société.

L’idée de fond de cet article est de montrer les problèmes les plus courants dans les infrastructures de synchronisation du temps basées sur les protocoles NTP et SNTP comme ils sont les plus largement implémentés dans les réseaux d’entreprise, mais les problèmes décrits dépendent un peu du niveau de protocole réseau. Si vous utilisez un protocole différent pour la synchronisation du temps, vous pouvez tout de même être intéressé.

I. Topologie de synchronisation réseau

La première question que vous devriez vous poser est comment l’information sur le temps est distribuée dans votre réseau, quelle est la topologie du service de synchronisation du temps et à quel point est fiable cette topologie.

La RFC 1305 définit le protocole NTP (Network Time Protocol) et décrit quelques modèles de synchronisation horaire différents. Elle contient un très bon examen de différents protocoles. Le protocole NTP supporte plusieurs modèles maîtres pour la synchronisation horaire. Ce modèle permet de créer partiellement (ou complètement) une topologie temporelle réseau maillée avec quelques, potentiellement externes, serveurs de temps. L’information reçue de tiers est filtrée pour fournir un résultat plus approprié au sens de statistiques mathématiques. L’information recueillie est utilisée pour calculer différentes mesures comme la déviance et l’exactitude et les paramètres d’horloge corrects. Si l’un des nœuds du réseau possède une configuration horaire invalide, cela n’aura pas d’impact sur le reste du réseau, puisque les méthodes de statistique permettent l’exclusion par filtrage des résultats invalides. Si il y a un hôte avec un horaire valide sur le réseau (c’est un hôte synchronisé avec une source matérielle quelconque, comme une horloge radio, horloge césium, horloge GPS, etc.) – l’ensemble du réseau sera finalement, après un certain temps, synchronisé avec cet hôte. Cette topologie maillée distribuée apporte une certaine sorte d’insertion à une infrastructure basée sur NTP. Si vous n’avez pas d’accès au réseau entier, vous ne pouvez pas faire changer l’heure au réseau entier immédiatement. Dans cette topologie, un serveur NTP externe public peut être utilisé comme source de temps de manière relativement sécurisée.

Le protocole NTP en lui-même n’est pas sécurisé – il est basé sur l’UDP, avec des ports source et destination fixes et est implémenté comme un simple service de paquets requête-réponse sans aucun nombre de séquence ou autre protection par répétition. Il permet facilement de falsifier aveuglement des paquets NTP (spoofing), mais pour réussir cette attaque vous devrez falsifier les réponses du grand nombre d’hôtes internes. Cela est difficile voir impossible sans avoir accès au réseau interne. Et, pour cause, cette topologie n’a pas un point unique d’erreur.

Pour mettre en place une certaine sorte de sécurité, NTP définit un authentificateur de message (message authenticator), pour permettre à un hôte de signer ses paquets NTP. Ce champ est optionnel et le mécanisme de signature de paquet, d’échange et de vérification de clé n’est pas défini par le protocole. Cela rend difficile d’implémenter cette protection dans un réseau d’entreprise et presque impossible d’implémenter l’authentificateur pour les serveurs de temps publics.

Comme d’habitude, le problème réside dans l’implémentation. Pour implémenter NTP dans une topologie partiellement maillée de plusieurs maîtres pour laquelle il fut inventé, vous devez configurer manuellement tous les tiers pour tous les tiers. Pour plus de sécurité (si les programmeurs l’ont implémentée) vous devez également pré échanger les clés ou mettre en place une infrastructure PKI. Après une configuration initiale, NTP ne requiert aucune maintenance. Mais, comme tout service configuré manuellement, il implique une charge de travail supplémentaire pour les administrateurs pour lire la documentation et mettre en place les paramètres initiaux. La plupart des administrateurs ne lise pas la documentation de NTP. C’est pourquoi, dans la plupart des cas, seulement un seul serveur NTP est configuré. Dans ce cas d’un seul maître, la topologie de synchronisation horaire devient une arborescence directe. Dans cette topologie, il n’y a plus de stabilité et de vérification d’erreur croisées. Si un hôte de plus haut niveau a des informations horaires invalides, un hôte synchronisé avec ce premier pourra ajuster son horloge sur une valeur intermédiaire. Il y a encore une certaine inactivité, mais cela ne protège plus votre réseau. La racine de l’arborescence est un point unique d’erreur et une bonne cible d’attaque.

Pour aider l'administrateur paresseux à conserver son temps réseau de la plus simple des manières, la RFC 1769 (Simple Network Time Protocol, SNTP) fut introduite. Le SNTP fonctionne en mode maître unique et les choses sont simples – vous synchronisez vos horloges avec un tiers distant, n'ayant que les décalages (lags) à prendre en compte. Pour cause, l'arborescence directe est une topologie uniquement disponible avec ce protocole. Pour la topologie d'arborescence, ce protocole est meilleur et n'est pas moins sécurisé. Au cas où les horloges du nœud principal (root node) sont changées, l'ensemble du réseau sera synchronisé dans un délai raisonnable (pour le NTP l'on peut rencontrer une situation où les différents niveaux de l'arborescence ont des horaires différents). Pour la sécurité des réseaux de communications SNTP est absolument le même – basé sur UDP, un paquet unique, ports source et destination tous deux en 123 et le statut du champ de l'Authentificateur n'est pas clair.

II. Implémentations

Quels sont les éléments nécessaires à une mise en place sécurisée de NTP ou SNTP? L'élément principal nécessaire est de supporter le champ Authentificateur au moins pour la synchronisation interne. Si l'Authentificateur n'est pas supporté, la seule alternative est d'utiliser IPSec pour signer tout le trafic de synchronisation (du fait que ce trafic ne contient pas d'informations sensibles devant être encryptées, le mode AH est préférable). Si l'Authentificateur est supporté, soit le PKI, soit la configuration manuelle d'une clé, soit les deux (et cela est préférable) doivent être supportés. Le support PKI est utile pour automatiser le déploiement d'une infrastructure de synchronisation horaire dans un réseau d'entreprise. La configuration manuelle d'une clé est nécessaire pour synchroniser avec un serveur de temps public (le serveur de temps public doit fournir une clé publique pour empêcher le spoofing). De ce fait, l'Authentificateur doit être implémenté afin d'empêcher les attaques par répétition (cela est requis par la RFC 1305).

Aussi, une implémentation sécurisée devrait fournir une configuration pour un délai maximum de correction. La RFC 1305 nécessite que ces paramètres soient modifiables par l'administrateur réseau. Le paramétrage du délai maximum de correction du temps sur une petite valeur peut causer des problèmes d'échec de synchronisation horaire d'un ordinateur client (par exemple si cet hôte est débranché pendant une longue période). Une grande valeur entraîne la possibilité d'attaques contre les services nécessitant une synchronisation horaire (voir le début de cet article), cela peut également conduire à une instabilité, du fait que la variation et la précision des horloges peuvent être calculées de manière invalide. En tant qu'architecte de l'infrastructure vous devrez trouver un juste milieu. Cette valeur pour les réseaux gérés est généralement inférieure à 15 minutes.

Même la possibilité de changer l'heure pour une certaine valeur relativement petite pendant une certaine période peut avoir un impact déplaisant sur la stabilité d'un service de synchronisation, car les valeurs de temps peuvent être malformées entraînant les valeurs pour la précision et variation des horloges à être incorrectement calculées sur un ordinateur synchronisé et résultant en une augmentation de la variation (dérive).

Une autre protection est la détection des doubles réponses. Une double réponse signifie généralement que quelqu'un est en train de tenter de falsifié la réponse d'un serveur NTP/SNTP. Il n'est pas évident la façon dont le système devrait réagir face à ce cas de double réponse (en dehors du fait que cette situation se doit absolument d'être enregistrée et que l'administrateur doit être alerté par tous les moyens possibles), la meilleure protection probable est d'ignorer les deux paquets, Mais cela pourrait rendre possible d'empêcher la synchronisation horaire par un flood de fausses réponses NTP/SNTP. Ce genre d'attaque est extrêmement simple mais peut nécessiter des mois avant d'obtenir de quelconques résultats négatifs.

Additionnelement, une fraction aléatoire doit être intégrée dans la planification de la synchronisation horaire pour rendre plus difficile la falsification transparente de réponse NTP/SNTP du serveur. Sans cette fraction, un attaquant distant peut de manière transparente falsifier la réponse serveur NTP avec un paquet unique.

III. Exemple: Forêt réseau Windows Active Directory

Le réseau Active directory Windows 2000 est choisi comme exemple d'infrastructure de synchronisation horaire pour plusieurs raisons : il est complètement implémenté et documenté par le vendeur, il requiert un minimum de configuration manuelle et il fait intervenir à la fois les problèmes et avantages que nous souhaitons traiter.

Avant Windows 2000, Microsoft n'avait pas d'infrastructure de synchronisation horaire. L'unique méthode disponible était la synchronisation manuelle (ou batchée) via la commande 'net time'. Cette commande nécessitait des privilèges élevés. Il était impossible de synchroniser le temps avec un script logon utilisateur sans donner une permission spéciale pour synchroniser l'horaire à l'utilisateur. Du fait que dans les réseaux Windows c'est à l'ordinateur client de vérifier les heures de connections permises, donner une possibilité à un utilisateur de changer l'heure n'est pas souhaitable.

Dans Windows 2000, Microsoft a fait un bon travail pour mettre en place une infrastructure de synchronisation du temps. Ceci est décrit en détails en [3]. Ce document décrit les impacts de sécurité dans la synchronisation horaire. L'approche de Microsoft sur la synchronisation horaire est très sérieuse. SNTP a été choisi comme transport. La synchronisation horaire supporte 3 modes : le mode Domaine (Nt5DS), le mode SNTP (NTP), et le mode sans synchronisation (NoSync) [6]. Dans le premier mode, un ordinateur membre du domaine se synchronise avec le contrôleur de domaine, le contrôleur de domaine se synchronise avec l'émulateur du PDC, l'émulateur du PDC se synchronise avec le contrôleur de domaine de la racine de la forêt du domaine, les contrôleurs de domaine dans les domaines de la racine de la forêt se synchronise avec l'émulateur du PDC de la racine du domaine de la forêt. Par défaut, dans Windows 2000, un émulateur PDC dans la racine d'une forêt de domaines n'est pas synchronisé et enregistre une erreur dans le journal d'applications. Dans ce mode, l'Authentificateur est employé et tous les paquets SNTP sont signés avec les clés de l'ordinateur. Pour la plupart des réseaux, cette topologie est suffisamment satisfaisante. C'est complètement automatisé et nécessite une administration minimale. C'est protégé contre les attaques de temps falsifié (pas contre les flood de paquets, mais cette attaque est moins significative).

Dans le dernier mode, un hôte ne synchronise pas avec une quelconque source externe et utilise seulement ses propres horloges (ce qui constitue dans la plupart des cas une source de temps valide pour le réseau).

Les problèmes apparaissent avec le mode SNTP. Dans ce mode, un hôte est synchronisé avec un serveur NTP externe (la RFC 1769 indique clairement que le SNTP ne doit pas être utilisé aux niveaux plus hauts que le niveau du sous réseau). La documentation Microsoft [3], recommande d'utiliser les sources de temps publiques pour synchroniser les horloges du premier contrôleur du domaine du premier domaine avec le temps absolu. Microsoft publie une liste des serveurs de temps publiquement disponibles [7]. Pour SNTP (synchronisation externe), aucune sécurité n'est supportée – pas de support d'Authentificateur/clé, pas de limitation pour les corrections horaire, pas de protection contre les paquets doubles et la planification de synchronisation horaire sans fraction aléatoire. Si le réseau est configuré en concordance avec ces recommandations, il est possible de mettre à genoux toute une forêt Windows 2003 avec un seul paquet UDP.

En 2001, j'avais contacté Microsoft au sujet de ce problème. Microsoft m'avait redirigé sur [3] (il était difficile de le trouver à l'origine) et m'avait recommandé d'utiliser IPSec comme indiqué dans les recommandations du [3]. Après ma réclamation qu'il est impossible d'utiliser IPSec avec des sources de temps publiques, Microsoft accepta ce problème comme un souci de conception sous la référence [MSRC 1088cb]. Pour aider l'utilisateur à trouver de la documentation, [4] et [5] furent publiés. Microsoft rapporta :

-=-= Citation du Centre de Réponse sur la Sécurité Microsoft =-=-

Nous sommes heureux de signaler que nous avons terminé tous les (nombreux, nombreux) correctifs et modifications suite au rapport que vous nous avez communiqué.

A ce niveau, nous sommes dans les tests finaux du Windows 2000 SP3, nous ne pouvons donc pas y intégrer des correctifs de cette ampleur pendant cette dernière étape : nous avons planifier d'inclure ceci dans le Windows 2000 SP4.

-=-= Fin de la Citation =-=-

En 2003, le SP4 fut diffusé. La documentation incluse à propos des correctifs ne parlait pas de SNTP. Ainsi, Windows XP et Windows 2003 furent diffusés avec un changement de configuration où time.windows.com n'est pas utilisé comme serveur de temps externe (SNTP) par défaut.

Initialement, je pensais que ce changement était fait pour permettre à Microsoft de signer les réponses depuis time.windows.com et pour vérifier sa clé sur l'hôte Windows et le même changement fut fait sur Windows 2000. Mais les tests démontrèrent que le serveur time.windows.com ne signe PAS les messages SNTP. Après l'impossibilité de trouver une quelconque information sur les correctifs SNTP incluent dans le SP4, je contactai Microsoft à nouveau en 2004. Après un dialogue avec le MSRC j'ai pu trouver quelques informations utiles, mais probablement incomplètes. Au moins, Microsoft ajouta une clé de registre non documentée pour le service de synchronisation du temps

MaxAllowedClockErrInSecs

Cette valeur permet de définir le délai de correction maximum pour une source externe (elle n'est jamais utilisée pour une synchronisation de domaine interne. Ce paramètre n'élimine pas le problème mais il est possible de l'utiliser sur un émulateur PDC d'une racine de forêt pour prévenir les services les plus importants d'une attaque immédiate, mais il est encore possible de lancer une attaque contre le calcul de divergence et précision du temps.

La valeur par défaut est 43200 (15 heures). 12 heures est une période immense qui permet à un grand nombre de services d'être attaqués. Je recommande d'utiliser une valeur inférieure à 60 sur un émulateur PDC de racine de forêt.

IV. Décisions de conception:

Pour les réseaux où la synchronisation horaire est critique pour assurer le service, considérez l'utilisation d'horloges matérielles (par exemple radio ou GPS) pour des sources valables. Implémentez une topologie maillée avec le protocole NTP pour les serveurs de distribution de l'heure de haut niveau et les hôtes à mission critique (dans un environnement Windows, vous devez utiliser un logiciel tiers et marquer ces hôtes comme sources de temps valables [5], [6] pour empêcher le service de temps Windows de corriger les horloges systèmes). Vérifiez que l'implémentation du NTP supporte le mécanisme d'Authentificateur. Si oui, échangez les clés (implémentez PKI si cela est nécessaire). Si l'implémentation de NTP ne supporte pas le mécanisme d'Authentificateur, ou si ce support ne correspond pas à vos besoins, mettez en place IPSec en mode AH entre les hôtes pour protéger le trafic de synchronisation. Les sources de temps fiables d'Internet ne peuvent être utilisées seulement pour des nécessités de moyens et de faible sécurité pour une topologie maillée avec NTP et seulement avec des nécessités de sécurité faible pour une topologie d'arborescence avec NTP ou SNTP.

Pour le reste du réseau, vous pouvez utiliser SNTP (par exemple la synchronisation horaire intégrée dans Windows) ou NTP en mode maître unique. La protection du trafic comme décrite ci-dessus est également requise.

Pour les protocoles NTP/SNTP, après la configuration initiale, il n'y a pas besoin d'administration régulière, mais un haut niveau de monitoring est requis. Chaque événement de synchronisation, spécialement ceux répétés pour quelques cycles de synchronisation doit être inspecté soigneusement sur une base journalière.

V. Références:

[1] David L. Mills, Network Time Protocol (Version 3) Specification, Implementation and Analysis, RFC 1305
http://ietf.org/rfc/rfc1305.txt
[2] D. Mills, Simple Network Time Protocol (SNTP), RFC 1769
http://ietf.org/rfc/rfc1769.txt
[3] The Windows Time Service
http://www.microsoft.com/windows2000/docs/wintimeserv.doc
[4] The Windows Time Service
http://www.microsoft.com/technet/prodtechnol/windows2000serv/maintain/operate/wintime.mspx
[5] Q224799 - Basic Operation of the Windows Time Service
http://support.microsoft.com/default.aspx?scid=kb;en-us;224799
[6] Q223184 - Registry Entries for the W32Time Service
http://support.microsoft.com/default.aspx?scid=kb;en-us;223184
[7] Q262680 - A List of the Simple Network Time Protocol Time Servers That Are Available on the Internet
http://support.microsoft.com/default.aspx?scid=kb;en-us;262680