Comprendre et dépanner la mise en oeuvre de Finesse BOSH

Table des matières

Introduction

Ce document décrit l'architecture derrière les connexions Finesse qui utilisent BOSH et comment les problèmes de connexion BOSH peuvent être diagnostiqués.

Conditions préalables

Exigences

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Cisco Finesse
- Unified Contact Center Enterprise (UCCE)
- Unified Contact Center Express (UCCX)
- Outils de développement de navigateur Web
- Administration de Windows et/ou Mac

Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Cisco Finesse 9.0(1) - 11.6(1)
- UCCX 10.0(1) - 11.6(2)

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre lâ€™incidence possible des commandes.

Informations générales

Les connexions qui utilisent des flux bidirectionnels sur HTTP synchrone sont appelées BOSH.

Comprendre la mise en oeuvre de Finesse BOSH

Comprendre XMPP

Le protocole XMPP (Extensible Messaging and Presence Protocol) (également connu sous le nom de Jabber) est un protocole avec état dans un modèle client-serveur. XMPP permet la livraison rapide de petits morceaux de données eXtensible Markup Language (XML) structurées d'une entité à une autre. XMPP/Jabber est largement utilisé dans les applications de messagerie instantanée et de présence.

Toutes les entités XMPP sont identifiées par leur ID Jabber (JID).

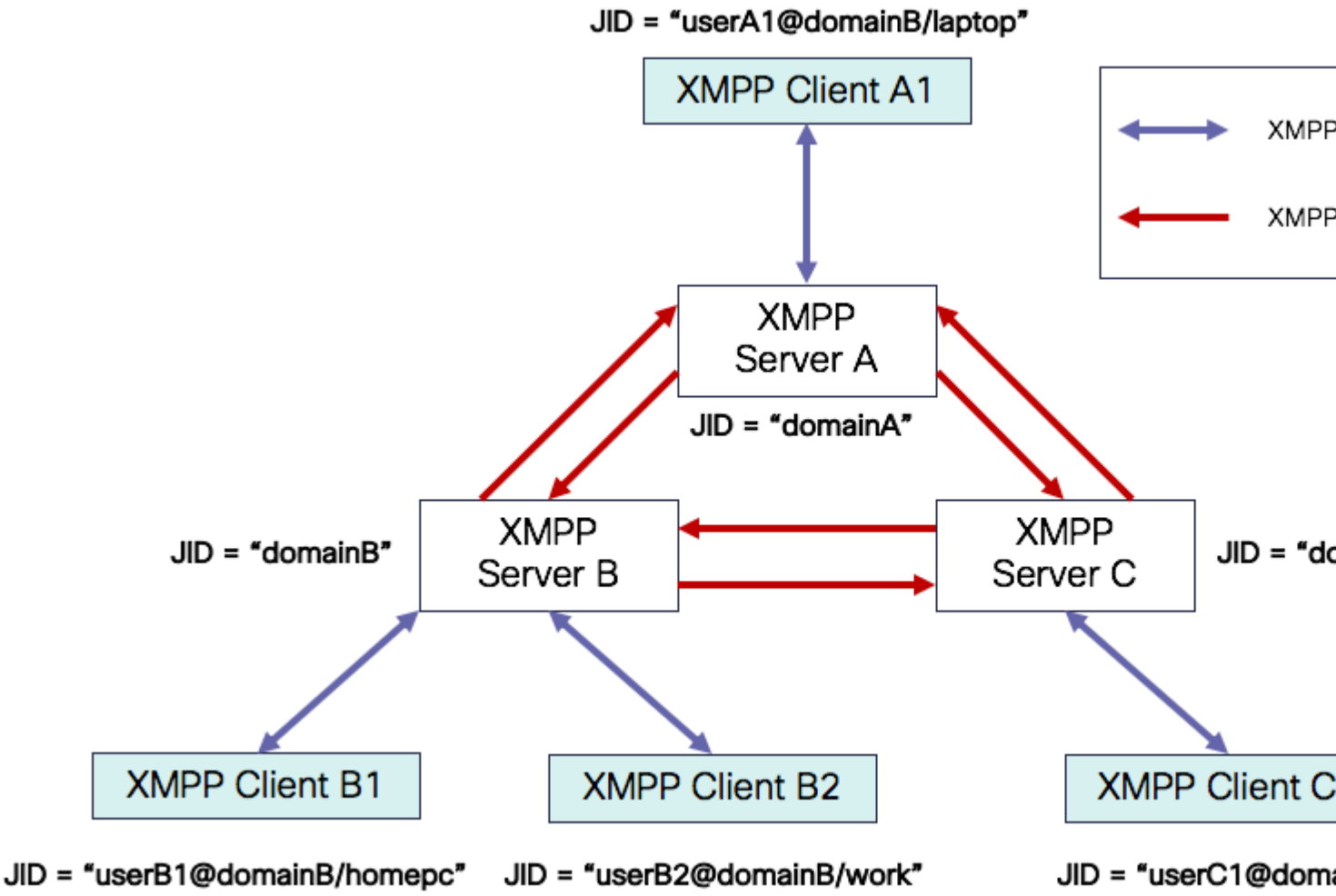


Schéma d'adressage JID : user@domain/resource

usager	nom d'utilisateur client sur le serveur XMPP ou nom de la salle de conférence
domaine	Nom de domaine complet (FQDN) du serveur XMPP
ressource	identifiant de l'entité/point d'extrémité spécifique de l'utilisateur (par exemple, ordinateur portable, smartphone, etc.), identifiant de session ou nom de noeud pubsub

---

> **Remarque** : les trois composants JID ne sont pas utilisés dans tous les cas. Un serveur est généralement défini par le domaine, une salle de conférence par user@domain et un client par user@domain/resource.

Les messages XMPP sont appelés strophes. Il y a trois strophes principales dans XMPP :

1. <message> : une direction, un destinataire

2. <présence> : une direction, publier sur plusieurs

3. <iq> : info/query - request/response

Toutes les strophes ont des adresses de destination et de provenance et la plupart des strophes ont également des attributs type, id et xml : langattribute.

Attribut Stanza	**Objectif**
par	JID de destination
expéditeur	JID source
type	objet du message
id	identifiant unique utilisé pour lier une requête à une réponse pour les <iq> stanzas
xml : lang	Définit la langue par défaut de tout code XML lisible par l'utilisateur dans la strophe

Exemple de message XMP

```
<message to='person1@example' from='person2@example' type='chat'>
  <subject> Team meeting </subject>
  <body>Hey, when is our meeting today? </body>
  <thread>A4567423</thread>
</message>
```

Mise en oeuvre XMPP avec finesse

Si une application Web doit fonctionner avec XMPP, plusieurs problèmes surviennent. Les navigateurs ne prennent pas en charge XMPP sur TCP (Transmission Control Protocol) en mode natif, de sorte que tout le trafic XMPP doit être géré par un programme qui s'exécute à l'intérieur du navigateur. Les serveurs et les navigateurs Web communiquent via des messages HTTP (HyperText Transfer Protocol), de sorte que Finesse et d'autres applications Web encapsulent les messages XMPP à l'intérieur des messages HTTP.

La première difficulté de cette approche est que HTTP est un protocole sans état. Cela signifie que chaque requête HTTP n'est liée à aucune autre requête. Cependant, ce problème peut être résolu par des moyens applicatifs, par exemple par l'utilisation de cookies ou de données postales.

La deuxième difficulté est le comportement unidirectionnel du protocole HTTP. Seul le client envoie des requêtes et le serveur ne peut que répondre. L'incapacité du serveur à transmettre des données rend anormale la mise en oeuvre de XMPP sur HTTP.

Ce problème n'existe pas dans la spécification XMPP Core d'origine (RFC 6120), où XMPP est lié à TCP. Cependant, si vous voulez résoudre le problème avec XMPP lié à HTTP, par exemple, parce que Javascript peut envoyer des requêtes HTTP, il y a deux solutions possibles. Les deux nécessitent un pont entre HTTP et XMPP.

Les solutions proposées sont les suivantes :

1. Interrogation (protocole hérité) : requêtes HTTP répétées demandant de nouvelles données définies dans XEP-0025 : Interrogation HTTP Jabber

2. L'interrogation longue est également appelée BOSH : protocole de transport qui émule la sémantique d'une connexion TCP bidirectionnelle à longue durée de vie entre deux entités en utilisant efficacement plusieurs paires requête/réponse HTTP synchrones sans nécessiter l'utilisation d'une interrogation fréquente définie dans XEP-0124 : HTTP Binding et étendue par XEP-0206 : XMPP sur BOSH

Finesse implémente BOSH car il est très efficace du point de vue de la charge du serveur et du trafic. La raison d'utiliser BOSH est de dissimuler le fait que le serveur n'a pas à répondre dès qu'il y a une demande. La réponse est retardée jusqu'à une durée spécifiée jusqu'à ce que le serveur dispose de données pour le client, puis elle est envoyée en tant que réponse. Dès que le client reçoit la réponse, il fait une nouvelle demande, etc.

Le client de bureau Finesse (application Web) établit une connexion BOSH périmée sur le port TCP 7443 toutes les 30 secondes. Au bout de 30 secondes, en l'absence de mises à jour du service de notification Finesse, le service de notification envoie une réponse HTTP avec un 200 OK et un corps de réponse (presque) vide. Si le service de notification dispose d'une mise à jour sur la présence d'un agent ou d'un événement de dialogue (appel), par exemple, les données sont envoyées immédiatement au client Web Finesse.

Exemple de requête/réponse XMPP Finesse

Cet exemple montre la première réponse de requête de message XMPP partagée entre le client Finesse et le serveur Finesse pour configurer la connexion BOSH.

```
Finesse client request:
<body xmlns="http://jabber.org/protocol/httpbind" xml:lang="en-US" xmlns:xmpp="urn:xmpp:xbosh" hold="1"

Finesse server response:
<body xmlns="http://jabber.org/protocol/httpbind" xmlns:stream="http://etherx.jabber.org/streams" authid
```

Pour récapituler :

1. Le client Web Finesse dispose d'une connexion HTTP périmée (http-bind) configurée sur le serveur Finesse via le port TCP 7443. C'est ce qu'on appelle un sondage long BOSH.

2. Le service de notification Finesse est un service de présence qui publie des mises à jour concernant l'état d'un agent, d'un appel, etc.

3. Si le service de notification dispose d'une mise à jour, il répond à la requête http-bind avec la mise à jour d'état sous la forme d'un message XMPP dans le corps de la réponse HTTP.

4. S'il n'y a aucune mise à jour d'état 30 secondes après la réception de la demande http-bind, le service de notification répond sans aucune mise à jour d'état pour permettre au client Web Finesse d'envoyer une autre demande http-bind. Cela permet au service de notification de savoir que le client Web Finesse est toujours en mesure de se connecter au service de notification et que l'agent n'a pas fermé son navigateur ou mis son ordinateur en veille, etc.

Comprendre les messages et les noeuds XMPP de Finesse

Finesse implémente également la spécification XMPP XEP-0060 : Publish-Subscribe. L'objectif de cette spécification est de permettre au serveur XMPP (service de notification) d'obtenir des informations publiées sur les noeuds XMPP (rubriques), puis d'envoyer des événements XMPP aux entités abonnées au noeud. Dans le cas de Finesse, le serveur CTI (Computer Telephony Integration) envoie des messages CTI au service Web Finesse pour informer Finesse des mises à jour de configuration telles que, mais sans s'y limiter, la création d'agent ou de file d'attente de service de contact (CSQ) ou des informations sur un appel. Ces informations sont ensuite converties en un message XMPP que le service Web Finesse publie sur le service de notification Finesse. Le service de notification Finesse envoie ensuite des messages XMPP sur BOSH aux agents qui sont abonnés à certains noeuds XMPP.

Certains des objets de l'API Finesse définis dans le Guide du développeur des services Web Finesse sont des noeuds XMPP. Les clients Web Finesse agent et superviseur peuvent s'abonner à des mises à jour d'événements pour certains de ces noeuds XMPP afin d'avoir des informations à jour sur les événements en temps réel (tels que les événements d'appel, les événements d'état, etc.). Ce tableau présente les noeuds XMPP qui sont activés pour pubsub.

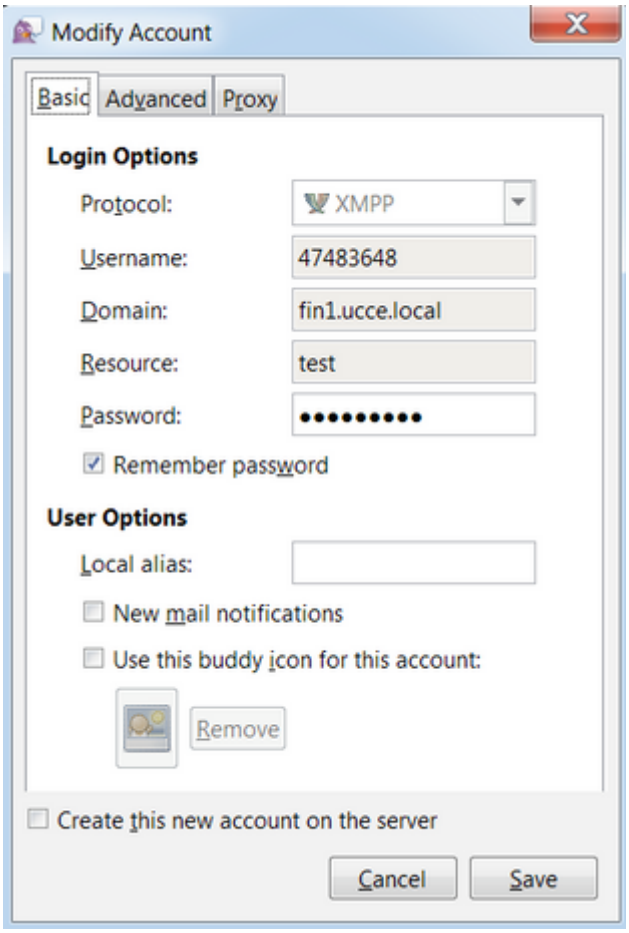
Finesse API, objet	**Objectif**	**Abonnement**
/finesse/api/User/<ID de connexion>	Affiche l'état et le mappage d'équipe de l'agent	Agents et superviseurs
/finesse/api/User/<ID de connexion>/Dialogues	Affiche les appels traités par l'agent	Agents et superviseurs
/finesse/api/User/<IdentifiantConnexion>/ClientLog	Utilisé pour capturer les journaux du client à partir du bouton **Envoyer le rapport d'erreur**	Agents et superviseurs
/finesse/api/User/<LoginID>/Queue/<queueID>	Affiche les données statistiques de la file d'attente (si activé)	Agents et superviseurs
/finesse/api/Team/<IDéquipe>/Users	Affiche les agents qui appartiennent à une certaine équipe, y compris les informations d'état	Superviseurs
/finesse/api/SystemInfo	Affiche l'état du serveur Finesse. Utilisé pour déterminer si le basculement est nécessaire	Agents et superviseurs

Exemple 1 : Utiliser Pidgin pour afficher les noeuds XMPP Finesse

Étape 1. Téléchargez et installez le Pidgin du client XMPP.

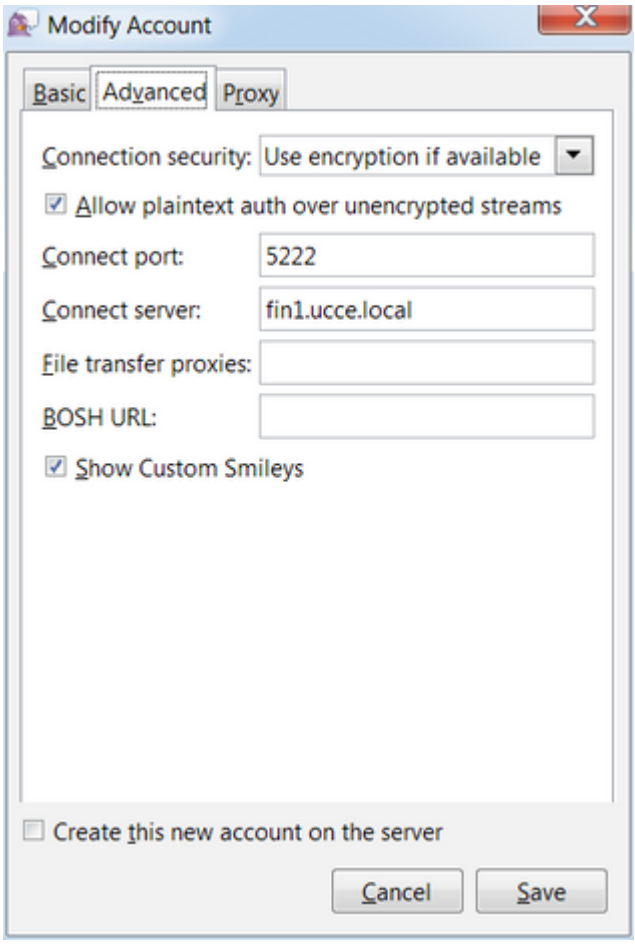
Étape 2. Accédez à **Accounts > Basic** et configurez les **Options de connexion** :

- Protocole : XMPP
- Nom d'utilisateur : LoginID pour tout agent
- Domaine : FQDN du serveur Finesse
- Ressource : Espace réservé - toute valeur peut être utilisée, par exemple, test
- Mot de passe : Agent password
- Cochez la case **Mémoriser le mot de passe**



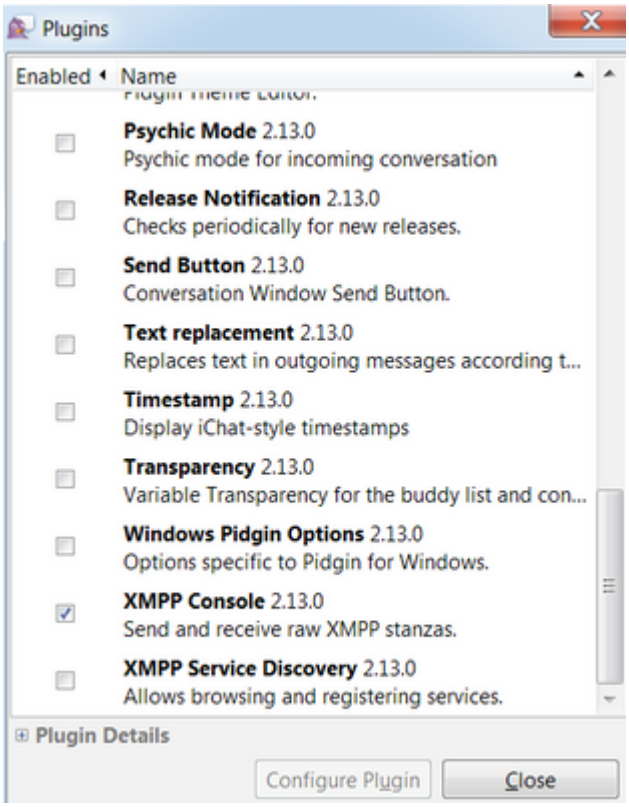
Étape 3. Accédez à **Comptes > Modifier > Avancé** et configurez :

- Sécurité de la connexion : utilisez le chiffrement si disponible
- Cochez la case **Allow plaintext auth other unencryption flows**.
- Port de connexion : 522. Utilisez le port par défaut 5222. Ce port est requis pour les clients XMPP externes. Les clients de bureau Finesse utilisent 7443. N'utilisez pas le port 7443.
- Serveur de connexion : FQDN du serveur Finesse

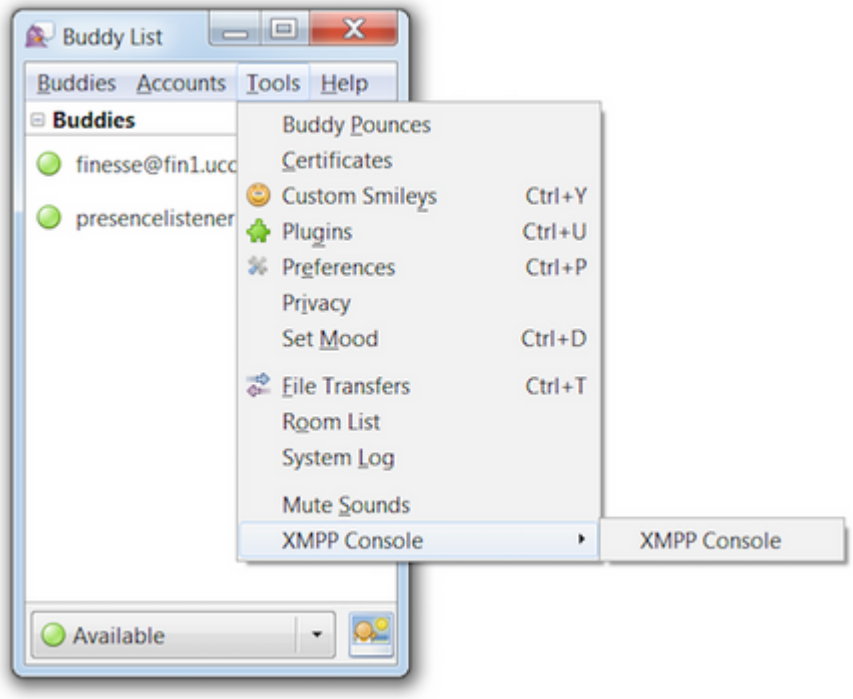


Remarque : le port 5222 est utilisé uniquement parce que les clients Web Finesse peuvent utiliser le port 7443 pour se connecter au service de notification.

Étape 4. Accédez à **Outils > Plugins** et activez la console XMPP.

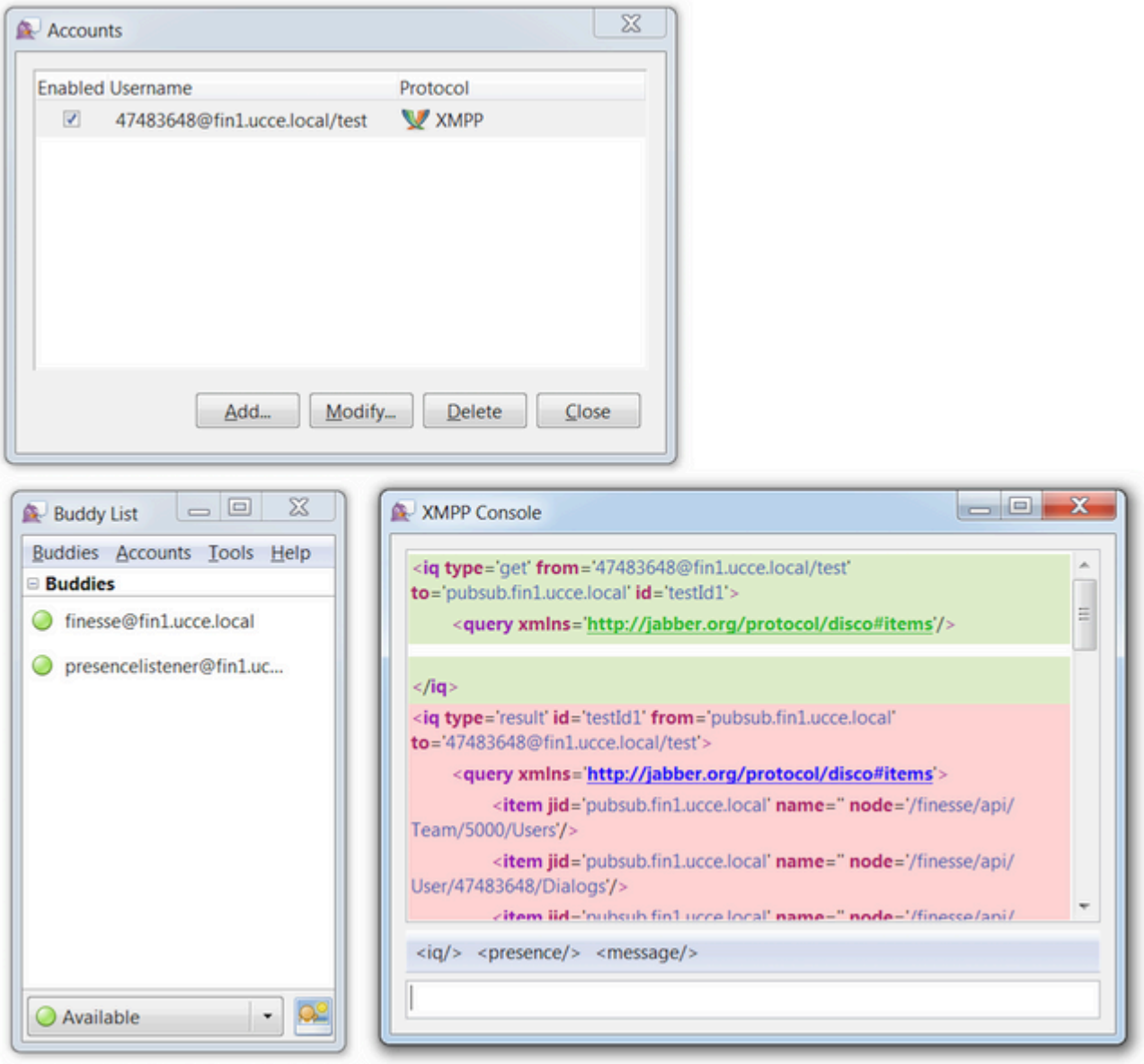


Étape 5. Accédez à **Outils > Console XMPP > Console XMPP** pour ouvrir la console XMPP.



Étape 6. Exécutez ce message **<iq>** pour voir tous les noeuds XMPP qui existent.

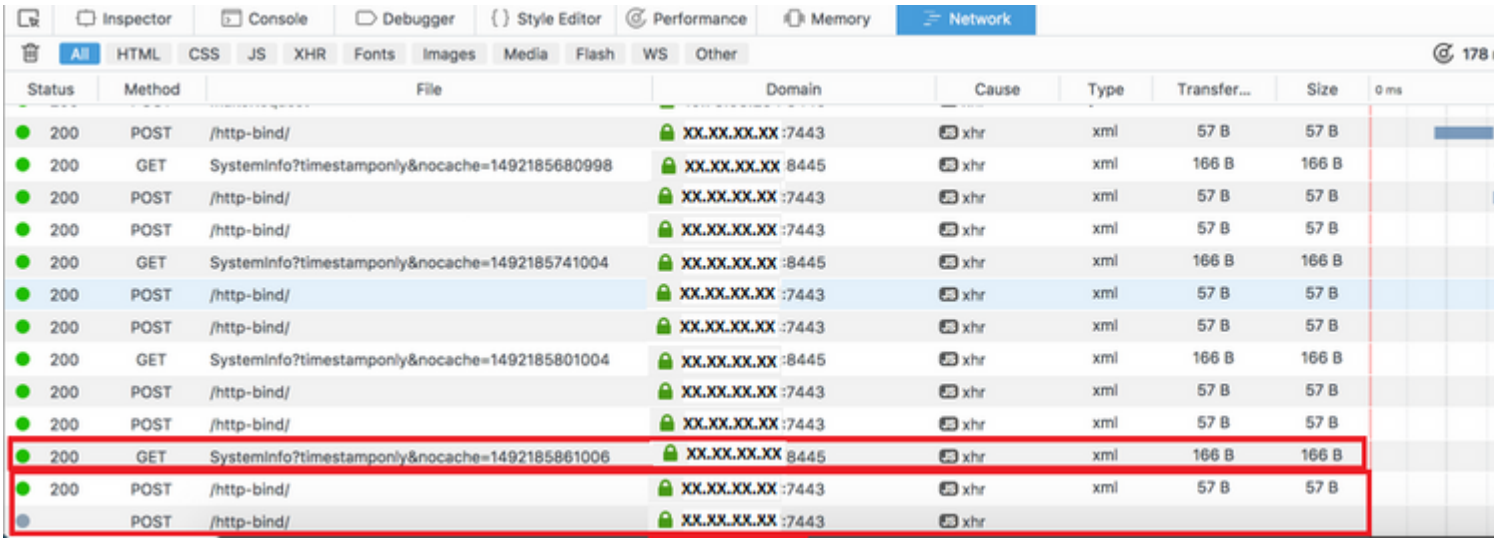
Exemple :



Dans un environnement de travaux pratiques avec deux agents et deux CSQ configurés, ce résultat est contenu dans la réponse Finesse :

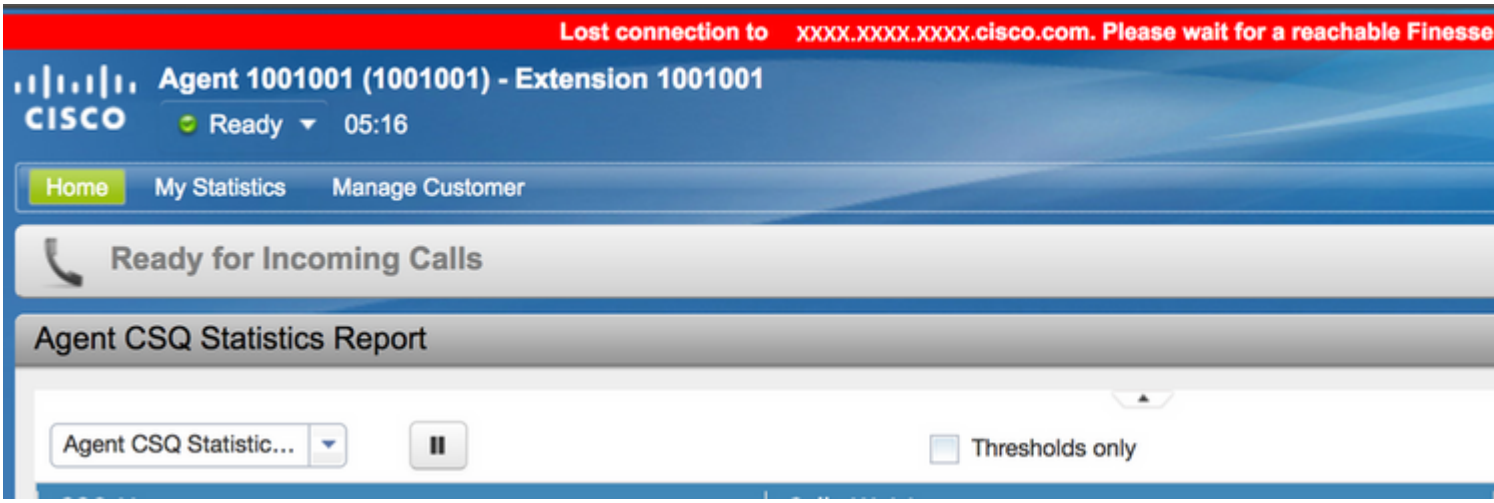
Exemple 2 : Utiliser l'onglet Réseau des outils de développement de navigateur pour afficher les messages HTTP

Chaque navigateur dispose d'un ensemble d'outils de développement. L'onglet Réseau des outils de développement affiche les messages HTTP envoyés et reçus par le client Web Finesse (navigateur). Par exemple, cette image montre comment le client Web Finesse envoie une requête SystemInfo qui vérifie l'état de Finesse Tomcat toutes les minutes en tant que vérification de basculement. En outre, les messages http-bind de la connexion BOSH sont également affichés. Le serveur Finesse renvoie une réponse dans les 30 secondes s'il n'y a aucune mise à jour à publier sur les noeuds XMPP auxquels le client Web est abonné.



Dépannage du message d'erreur de déconnexion BOSH

Lorsqu'une déconnexion BOSH se produit, l'erreur Connexion perdue à {Finesse Server FQDN}. Veuillez attendre qu'un serveur Finesse accessible soit trouvé... s'affiche dans une bannière rouge en haut du bureau Finesse.



Ce message s'affiche car aucun événement d'abonnement XMPP ne peut actuellement être reçu du service de notification Cisco Finesse. Par conséquent, les informations d'état et les détails d'appel ne peuvent pas être affichés sur le bureau de l'agent.

Pour UCCX, 60 secondes après la déconnexion du navigateur, l'agent passe à l'état Déconnexion. L'agent peut être à l'état Prêt ou Non prêt pour que la déconnexion se produise.

Pour UCCE, Finesse prend jusqu'à 120 secondes pour détecter quand un agent ferme le navigateur ou que le navigateur tombe en panne et Finesse attend 60 secondes avant d'envoyer une demande de déconnexion forcée au serveur CTI, ce qui amène le serveur CTI à mettre l'agent dans un état Non prêt. Dans ces conditions, Finesse peut prendre jusqu'à 180 secondes pour déconnecter l'agent. Contrairement à UCCX, l'agent passe à l'état Non prêt au lieu de l'état Déconnexion.

> **Remarque** : la déconnexion CTI n'est pas prête et Le comportement de l'état de déconnexion dans UCCE est contrôlé par le paramètre PG /LOAD. Selon les Notes de version de Unified Contact Center Enterprise et Hosted version 10.0(1), le paramètre /LOAD est déconseillé à partir de UCCE 10.0.

Pour plus d'informations sur le comportement d'UCCE Finesse Desktop, reportez-vous à la section Desktop Behavior du chapitre Cisco Finesse Failover Mechanisms du Guide d'administration de Cisco Finesse.

Remarque : les valeurs du minuteur peuvent changer à l'avenir selon les exigences du produit.

Analyse des journaux

Les journaux de service de notification Finesse et UCCX peuvent être collectés via RTMT ou via l'interface de ligne de commande :

fichier get activelog /desktop recurs compress

Journaux du service de notification de débogage

Remarque : définissez les journaux de niveau de débogage uniquement lorsque vous reproduisez un problème. Désactivez les débogages une fois le problème reproduit.

Remarque : Finesse 9.0(1) n'a pas de journalisation de niveau de débogage. La journalisation au niveau du débogage a été introduite dans Finesse 9.1(1). Le processus d'activation de la journalisation est différent dans 9.1(1) par rapport à Finesse 10.0(1) - 11.6(1). Pour ce processus, consultez le guide Finesse Administration and Serviceability.

Activez les journaux de débogage du service de notification de Unified Contact Center Express (UCCX), comme indiqué :

<#root>

admin:

utils uccx notification-service log enable

WARNING! Enabling Cisco Unified CCX Notification Service logging can affect system performance and should be disabled when logging is not required.

Do you want to proceed (yes/no)? yes

Cisco Unified CCX Notification Service logging enabled successfully.

NOTE: Logging can be disabled automatically if Cisco Unified CCX Notification Service is restarted.

Activez les journaux de débogage du service de notification de Unified Contact Center Enterprise (UCCE) (autonome Finesse), comme indiqué :

<#root>

admin:

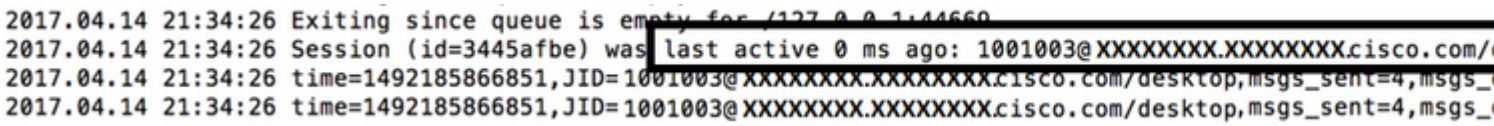
utils finesse notification logging enable

```
Checking that the Cisco Finesse Notification Service is started...
The Cisco Finesse Notification Service is started.

Cisco Finesse Notification Service logging is now enabled.

WARNING! Cisco Finesse Notification Service logging can affect system performance
and should be disabled when logging is not required.

Note: Logging can be disabled automatically if you restart the Cisco Finesse Notification Service
```

Ces journaux se trouvent dans le dossier /desktop/logs/openfire et sont nommés debug.log.

Comme l'illustre l'image, le fichier debug.log du service de notification (Openfire) affiche la liaison http avec le bureau, ainsi que l'adresse IP et le port du PC agent.

Comme lâ€™illustre lâ€™image, la dernière durée active de 0 ms indique que la session est toujours active.



Openfire fermant la session inactive indique que la déconnexion de l'agent peut se déclencher en 60 secondes, où Finesse peut envoyer une déconnexion forcée avec un code de raison de 255 au serveur CTI. Le comportement réel du bureau dans ces conditions dépend du paramètre de déconnexion lors de la déconnexion de l'agent (LOAD) dans UCCE. Dans UCCX, c'est toujours le comportement.

Si le client Finesse n'envoie pas de messages http-bind au serveur Finesse, les journaux peuvent afficher le temps de fonctionnement de la session et indiquer la fermeture de la session.

```
2017.06.17 00:14:34 Session (id=f382a015) was last active 0 ms ago: 1001003@xxxxx.xxxx.xxx.cisco.com/des
2017.06.17 00:15:04 Session (id=f382a015) was last active 13230 ms ago: 1001003@xxxxx.xxxx.xxx.cisco.com
2017.06.17 00:15:34 Session (id=f382a015) was last active 43230 ms ago: 1001003@xxxxx.xxxx.xxx.cisco.com
2017.06.17 00:16:04 Session (id=f382a015) was last active 63231 ms ago: 1001003@xxxxx.xxxx.xxx.cisco.com

2017.06.17 00:17:04 Unable to route packet. No session is available so store offline. <message from="pub
```

Journaux du service de notification Info

Ces journaux se trouvent dans le dossier /desktop/logs/openfire et sont nommés info.log. Si le client Finesse n'envoie pas de messages http-bind au serveur Finesse, les journaux peuvent indiquer que la session est devenue inactive.

```
2017.06.17 00:16:04 Closing idle session (id=f382a015): 1001003@xxxxx.xxxx.xxx. cisco.com/desktop
after inactivity for more than threshold value of 60
2017.06.17 00:16:04 A session is closed for 1001003@xxxxx.xxxx.xxx. cisco.com/desktop
```

Journaux des services Web

Ces journaux se trouvent dans le dossier /desktop/logs/webservices et sont nommés Desktop-webservices.YYY-MM-DDTHH-MM-SS.sss.log. Si le client Finesse n'envoie pas de messages http-bind au serveur Finesse dans le délai spécifié, les journaux peuvent afficher l'état d'indisponibilité de la présence de l'agent et, 60 secondes plus tard, une déconnexion pilotée par la présence peut se produire.

```
0000001043: XX.XX.XX.XXX: Jun 17 2017 00:16:04.630 +0530: %CCBU_Smack Listener Processor (1)-6-PRESENCE_
0000000417: XX.XX.XX.XXX: Jun 17 2017 00:16:04.631 +0530: %CCBU_Smack Listener Processor (1)-6-UNSUBSCRI
0000001044: XX.XX.XX.XXX:  Jun 17 2017 00:16:04.631 +0530: %CCBU_Smack Listener Processor (1)-6-AGENT_PR
0000001051: XX.XX.XX.XXX:  Jun 17 2017 00:16:35.384 +0530: %CCBU_pool-8-thread-1-6-AGENT_PRESENCE_MONITO
0000001060: XX.XX.XX.XXX:: Jun 17 2017 00:17:04.632 +0530: %CCBU_CoreImpl-worker12-6-PRESENCE DRIVEN LOG
0000001061: XX.XX.XX.XXX:: Jun 17 2017 00:17:04.633 +0530: %CCBU_CoreImpl-worker12-6-MESSAGE_TO_CTI_SERV
1, workmode : 0, reason code: 255, forceflag :1, agentcapacity: 1, agentext: 1001003, agentid: 1001003,
0000001066:  XX.XX.XX.XXX:: Jun 17 2017 00:17:04.643 +0530: %CCBU_CTIMessageEventExecutor-0-6-DECODED_ME
skillGroupNumber=-1, skillGroupPriority=0, agentState=1 (LOGOUT), eventReasonCode=255, numFltSkillGroups
duration=null, nextAgentState=null, fltSkillGroupNumberList=[], fltSkillGroupIDList=[], fltSkillGroupPri
msgID=30, timeTracker={"id":"AgentStateEvent","CTI_MSG_RECEIVED":1497638824642,"CTI_MSG_DISPATCH":149763
Decoded Message to Finesse from backend cti server
```

Raisons courantes de la déconnexion BOSH

Les connexions BOSH sont configurées par le client Web et le serveur Finesse détermine si la présence de l'agent n'est pas disponible. Ces problèmes sont presque toujours des problèmes côté client liés au navigateur, à l'ordinateur de l'agent ou au réseau, car la responsabilité du démarrage de la connexion incombe au client.

Problème - Déconnexion des agents à différents moments (problème côté client)

Actions recommandées

Recherchez les problèmes suivants :

1. Problème réseau :

- Examiner les règles et les journaux du pare-feu : le port TCP 7443 ne doit pas être bloqué ou limité
- Utilisez un analyseur de trafic Web HTTP tel que Fiddler® ou Wireshark® pour confirmer que le navigateur envoie des requêtes http-bind sur le port TCP 7443 et reçoit des réponses
- Vérifier tous les périphériques/interfaces réseau entre l'ordinateur de l'agent et le serveur Finesse pour

détecter tout retard excessif ou toute perte de paquets
    - La commande traceroute peut être utile pour déterminer le chemin et les délais
        - Sur un PC Microsoft® Windows® : tracert {Finesse Server IP | Nom de domaine complet du serveur Finesse}
        - Sur un Mac® : traceroute {Finesse Server IP | Nom de domaine complet du serveur Finesse}
        - Sur le logiciel Cisco IOS®, les statistiques d'interface peuvent être vérifiées : show interfaces
            - Référez-vous Dépannage des pertes de file d'attente en entrée et de sortie

- Collecter les journaux du client Finesse pour un agent de test. Les journaux clients peuvent être collectés de trois manières :
    1. Journaux de console Web du navigateur
        - Console Web Firefox
        - Console Web Microsoft Edge
        - Console Web Chrome
    2. Appuyez sur le bouton Send Error Report sur la page Finesse et collectez les journaux du serveur Finesse. Les journaux se trouvent dans /desktop/logs/clientlogs.
    3. Connectez-vous via https://<Finesse-FQDN>/desktop/locallog et collectez les journaux après le problème.

Chaque minute, le client se connecte au serveur Finesse pour calculer la dérive et la latence du réseau :

```
<PC date-time with GMT offset>: : <Finesse FQDN>: <Finesse server date-time with offset>:
Header : Client: <date-time>, Server: <date-time>, Drift: <drift> ms, Network Latency (round trip): <RTT

2019-01-11T12:24:14.586 -05:00: : fin1.ucce.local: Jan 11 2019 11:24:14.577 -0600: Header : Client: 2019
```

En cas de problèmes de collecte de journaux, référez-vous à Dépannage du problème de journalisation persistante de Cisco Finesse Desktop

2. Navigateur et/ou version non pris en charge :

Utiliser le navigateur/la version et les paramètres pris en charge conformément aux matrices de compatibilité :

Matrice de compatibilité UCCE

Matrice de compatibilité UCCX

3. État de blocage du navigateur en raison du contenu/traitement d'un autre onglet/fenêtre :

Vérifiez le workflow de l'agent pour voir s'il :

- Ont généralement d'autres onglets ou fenêtres en cours d'exécution qui exécutent constamment d'autres applications en temps réel telles que la musique/vidéo en continu, les connexions WebSocket, les clients Web personnalisés de gestion de la relation client (CRM), etc
- Ouvrir un très grand nombre d'onglets ou de fenêtres
- Ont désactivé la mise en cache du navigateur
- ont maintenu leur navigateur en fonctionnement pendant une longue période et ne ferment pas le navigateur à la fin de la journée de travail

4. Ordinateur mis en veille :

Vérifiez si l'agent met son ordinateur en veille avant de se déconnecter de Finesse ou si le minuteur de mise en veille de son ordinateur est très bas.

5. Problème de CPU élevé ou de mémoire élevée sur l'ordinateur client :

- Si le navigateur de l'agent s'exécute dans un environnement partagé tel que Microsoft Windows Remote Desktop Services, Citrix® XenApp®, Citrix XenDesktop®, déterminez si les performances du navigateur dépendent du nombre d'utilisateurs exécutant le navigateur en même temps
    - Assurez-vous que la mémoire et les ressources processeur appropriées sont configurées en fonction du nombre d'utilisateurs

- Vérifier les problèmes d'utilisation des ressources informatiques :
    - Fenêtres:
        - Commande Windows PowerShell Get-Counter qui vérifie le % de temps processeur, le mégaoctet de mémoire disponible et le % de mémoire utilisée toutes les 2 secondes : Get-Counter -Counter "\Processor(_Total)\% Processor Time","\Memory\Available MBytes","\Memory\% Committed Bytes In Use" -SampleInterval 2 -Continuous
        - Au lieu d'utiliser PowerShell pour afficher les compteurs de performance de Windows, Windows Performance Monitor peut être utilisé
        - Le Gestionnaire des tâches peut être utilisé pour afficher les statistiques de CPU et de mémoire en direct globalement et processus par processus
    - Mac :
        - Commande Terminal Top qui vérifie le CPU total et la mémoire en direct : top
            - Vérifier les processus et trier par utilisation du CPU : top -o CPU
            - Vérifier les processus et trier par utilisation de la mémoire : top -o MEM
        - Activity Monitor peut être utilisé pour afficher les statistiques de CPU et de mémoire en direct globalement et processus par processus

6. Gadgets tiers effectuant une activité inattendue et problématique en arrière-plan :

Testez le comportement du bureau Finesse en supprimant tous les gadgets tiers.

7. Problème NTP sur le serveur ou le client :

- Vérifiez l'**état ntp utils** sur le serveur de publication Finesse pour vous assurer que la strate de serveur NTP est 4 ou inférieure
- Dans les journaux du client, vérifiez la dérive et la latence du réseau

Problème : tous les agents se déconnectent simultanément (problème côté serveur)

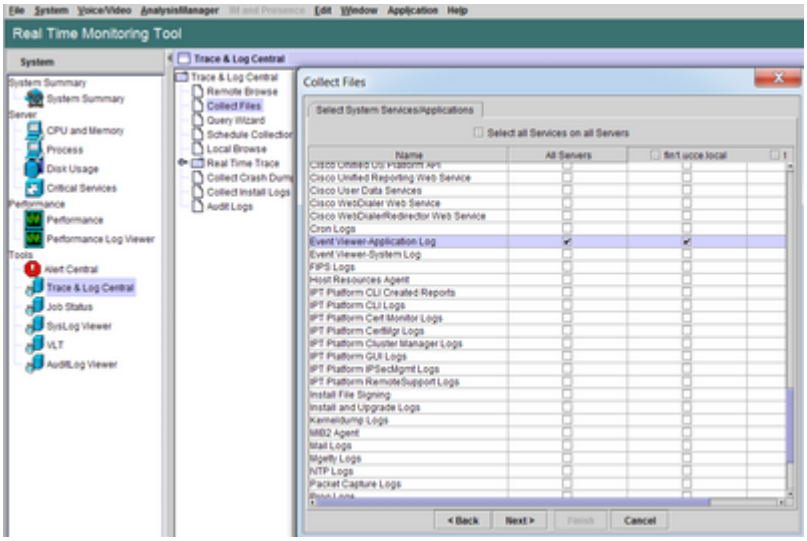
Actions recommandées

Recherchez les problèmes suivants :

1. Déconnexion du service Cisco Unified Communications Manager CTIManager. Si tous les fournisseurs CTIManager pour UCCX sont arrêtés ou se bloquent, les agents UCCX voient l'erreur de bannière rouge. Les agents UCCE ne voient pas la bannière rouge si cela se produit, mais les appels ne parviennent pas à être acheminés correctement vers les agents.

- Vérifiez si le service Cisco TIManager est démarré sur les serveurs CUCM utilisés comme fournisseurs CTI
- Vérifiez si le service Cisco CTIManager s'est bloqué via l'Observateur d'événements - L'application se

connecte à RTMT pour voir si le service Cisco CTIManager s'est bloqué
    - Pour collecter les journaux de l'Observateur d'événements sur RTMT, accédez à **Système > Outils > Trace et Log Central > Collecter les fichiers > Sélectionner les services/applications du système > Journal des applications de l'Observateur d'événements**.



- Pour collecter les journaux d'application de l'Observateur d'événements sur l'interface de ligne de commande : fichier get activelog /syslog/CiscoSyslog* abstime hh:mm:MM/DD/YY hh:mm:MM/DD/YY
- Pour afficher les vidages principaux sur l'interface de ligne de commande : utilise la liste active principale

---

Remarque : les noms de fichier des vidages principaux utilisent le format : core.<ProcessID>.<SignalNumber>.<ProcessName>.<EpochTime>.
Exemple : core.24587.6.CTIManager.1533441238
Par conséquent, l'heure de l'accident peut être déterminée à partir de l'heure de l'époque.

---

2. Arrêt ou blocage du service de notification Finesse/UCCX :

- Recherchez les erreurs du service de notification dans les journaux d'application de l'Observateur d'événements ou vérifiez si le service a été arrêté
- Vérifiez si le service de notification est activé : utils service list
- Vérifiez les heures d'arrêt du service de notification : file search activelog /desktop/logs/openfire "Openfire arrêté"
- Vérifiez les heures de démarrage du service de notification : file search activelog /desktop/logs/openfire "HTTP bind service started"
- Recherchez les vidages de mémoire du service de notification résultant d'une panne : file list activelog /desktop/logs/openfire/*.hprof
- Vérifiez si le service de notification écoute le trafic sur le port TCP 7443 : show open ports regexp 7443.*LISTEN
- Vérifiez si ces défauts sont applicables (ces défauts provoqueraient un échec de connexion pour les agents qui se connectent et pour les agents déjà connectés, ces agents verraient le message de déconnexion Finesse de la bannière rouge) :
    - ID de bogue Cisco CSCva7280 - Tomcat Finesse et crash d'Openfire pour caractères XML non valides
    - ID de bogue Cisco CSCva72325 - UCCX : Finesse Tomcat et Openfire Crash pour caractères XML non valides

Redémarrez Cisco Finesse Tomcat et le service de notification si vous suspectez une panne. Ceci n'est recommandé que dans une situation de réseau en panne, sinon, ces redémarres déconnectent les agents du serveur Finesse.

Étapes pour UCCE :

- utils service stop Cisco Finesse Tomcat
- utils service stop Service de notification Cisco Finesse
- utils service start Cisco Finesse Tomcat
- utils service start Service de notification Cisco Finesse

Étapes pour UCCX :

- utils service stop Cisco Finesse Tomcat
- utils service stop Service de notification Cisco Unified CCX
- utils service start Cisco Finesse Tomcat
- utils service start Service de notification Cisco Unified CCX

Utiliser Fiddler

La configuration de Fiddler peut être une tâche assez difficile sans comprendre les étapes nécessaires et le fonctionnement de Fiddler. Fiddler est un proxy web man-in-the-middle qui se trouve entre le client Finesse (navigateur web) et le serveur Finesse. En raison des connexions sécurisées entre le client Finesse et le serveur Finesse, cela ajoute une couche de complexité à la configuration de Fiddler afin de visualiser les messages sécurisés.

Émission De Violon Commun

Fiddler se trouvant entre le client Finesse et le serveur Finesse, l'application Fiddler doit créer des certificats signés pour tous les ports TCP Finesse qui nécessitent des certificats :

Certificats de service Cisco Finesse Tomcat

1. Serveur d'édition Finesse TCP 8445 (et/ou 443 pour UCCE)
2. Serveur d'abonnés Finesse TCP 8445 (et/ou 443 pour UCCE)

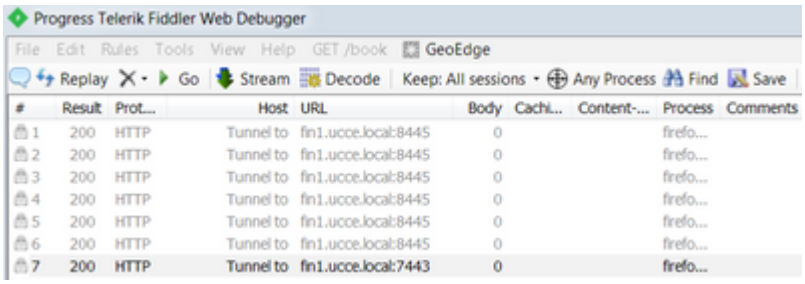
Certificats du service de notification Cisco Finesse (Unified CCX)

1. Serveur d'édition Finesse TCP 7443
2. Serveur d'abonnés Finesse TCP 7443

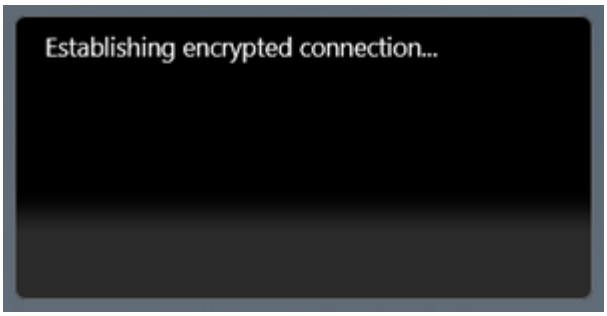
Le déchiffrement HTTPS doit être activé pour que Fiddler puisse générer dynamiquement des certificats pour le compte du serveur Finesse. Cette option n'est pas activée par défaut.

Si le déchiffrement HTTPS n'est pas configuré, la connexion initiale du tunnel au service de notification est visible, mais le trafic http-bind ne l'est pas. Fiddler ne montre que :

```
Tunnel to <Finesse server FQDN>:7443
```



Ensuite, les certificats Finesse signés par Fiddler doivent être approuvés par le client. Si ces certificats ne sont pas approuvés, il est impossible de passer l'étape Établissement d'une connexion chiffrée... de la connexion Finesse.



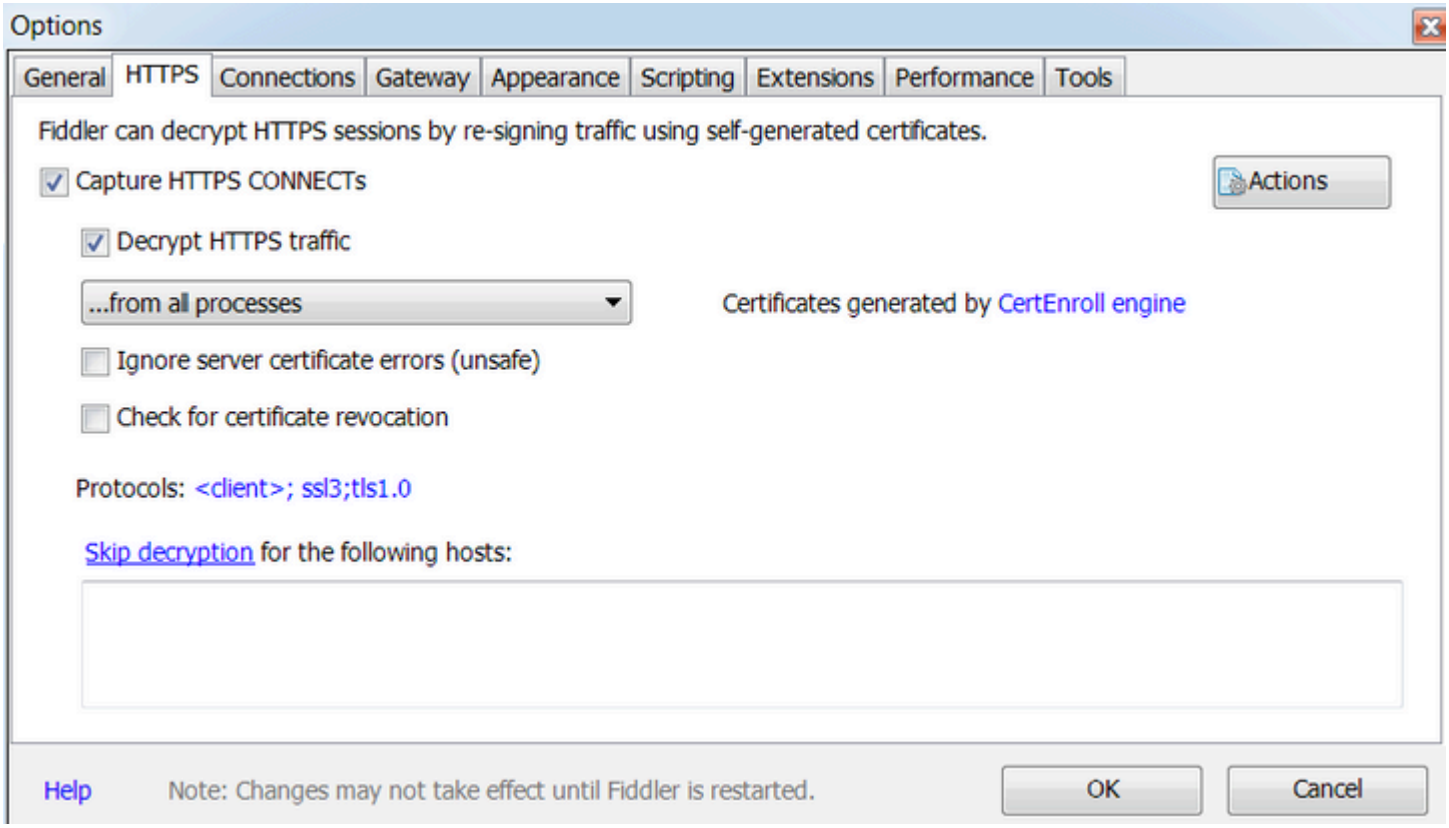
Dans certains cas, l'acceptation des exceptions de certificat à partir de la connexion ne fonctionne pas, et les certificats doivent être approuvés manuellement par le navigateur.

Exemple d'étapes de configuration

> **Attention** : l'exemple de configuration fourni concerne Fiddler v5.0.20182.28034 pour .NET 4.5 et Mozilla Firefox 64.0.2 (32 bits) sous Windows 7 x64 dans un environnement de travaux pratiques. Ces procédures ne peuvent pas être généralisées à toutes les versions de Fiddler, à tous les navigateurs ou à tous les systèmes d'exploitation des ordinateurs. Si votre réseau est actif, assurez-vous de comprendre l'impact potentiel de toute configuration. Consultez la documentation officielle de Fiddler pour plus d'informations.

Étape 1. Télécharger Fiddler

Étape 2. Activez le déchiffrement HTTPS. Accédez à **Tools > Options > HTTPS** et cochez la case **Decrypt HTTPS traffic**.

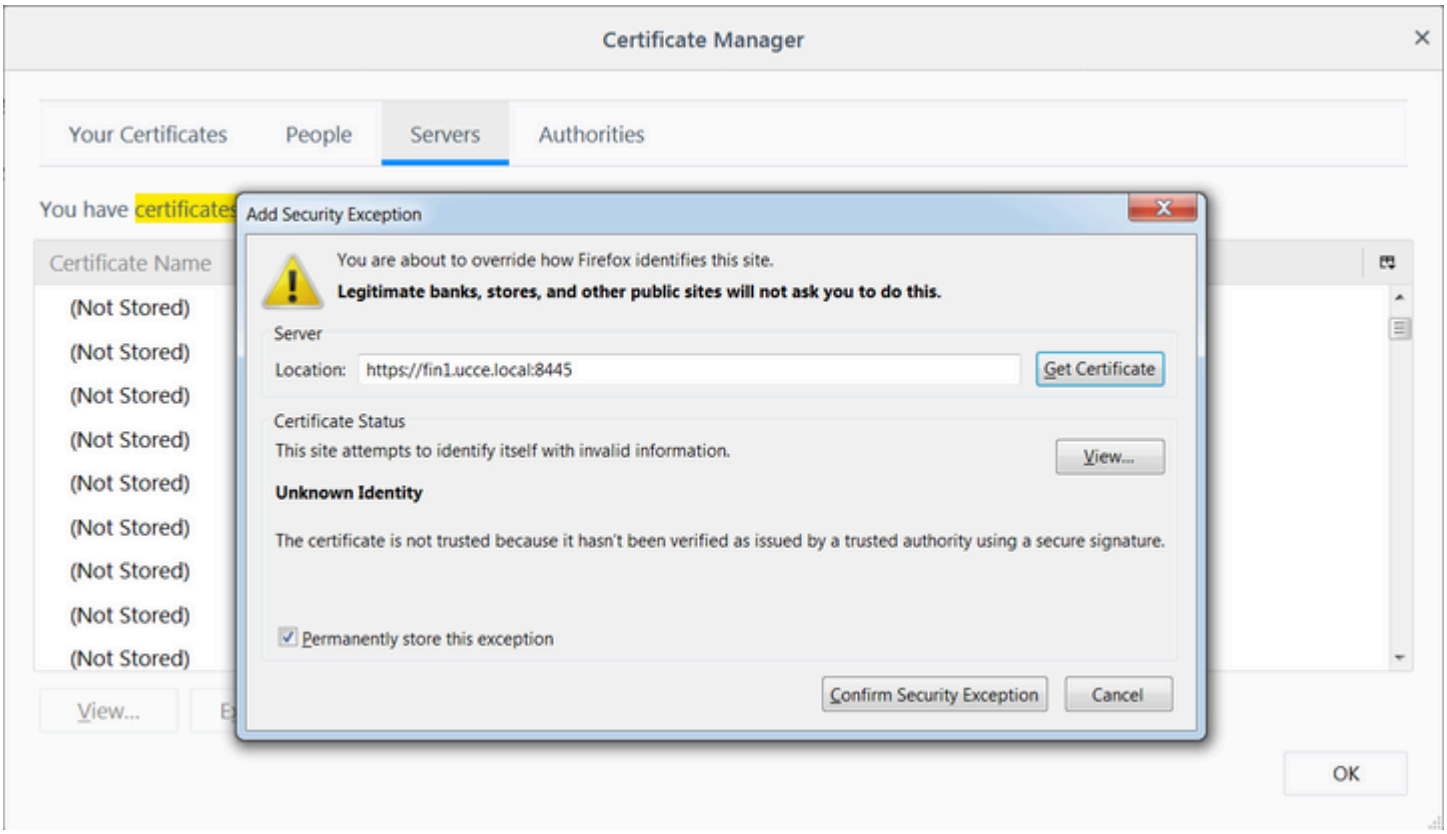


Étape 3. Un message d'avertissement s'affiche pour demander l'approbation du certificat racine Fiddler. Sélectionnez **Oui**.

Étape 4. Un message d'avertissement s'affiche avec le message « Vous êtes sur le point d'installer un certificat d'une autorité de certification (AC) prétendant représenter : DO_NOT_TRUST_FiddlerRoot... Voulez-vous installer ce certificat ?". Sélectionnez **Oui**.

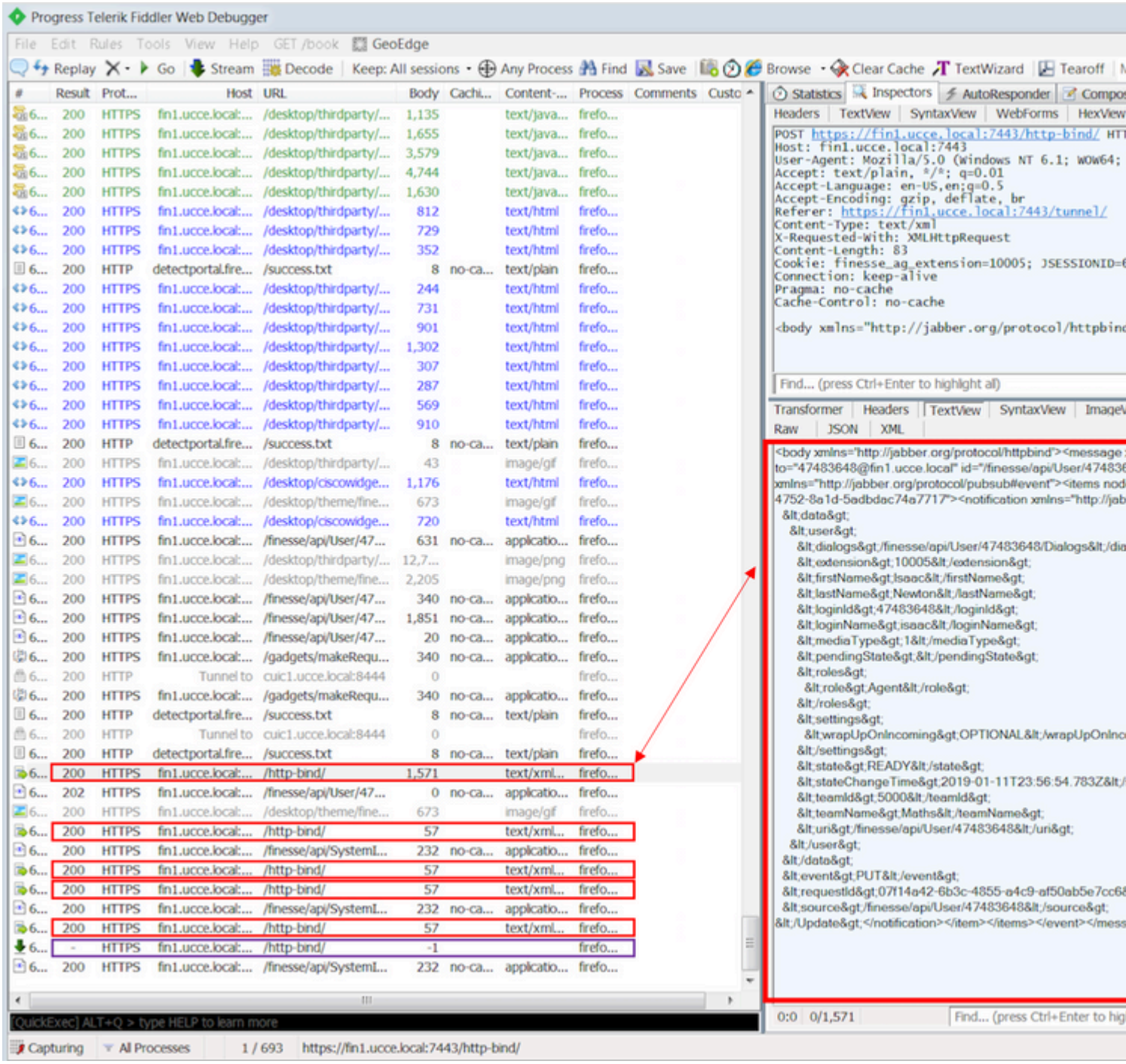
Étape 5. Ajoutez manuellement les certificats d'éditeur et d'abonné Finesse au magasin de certificats de confiance de l'ordinateur ou du navigateur. Assurez-vous que les ports 8445, 7443 et (uniquement pour UCCE) 443. Par exemple, sur Firefox, cela peut être fait simplement sans télécharger de certificats à partir de la page Finesse Operating System Administration :

Options > Find in Options (search) **> Certificates > Servers > Add Exception > Location >** Enter https://<Finesse server>:port for the relevant ports for both Finesse servers.

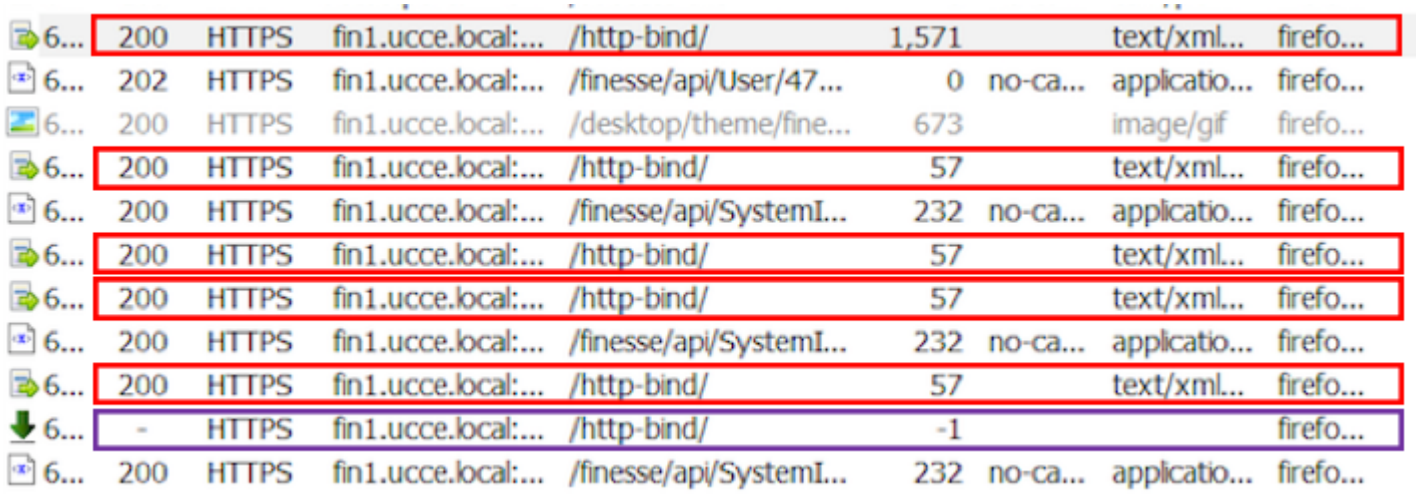


Étape 6. Connectez-vous à Finesse et voyez les messages http-bind qui laissent le client Finesse au serveur Finesse via Fiddler.

Dans l'exemple fourni, les 5 premiers messages affichent des messages http-bind auxquels le serveur Finesse a répondu. Le premier message contient 1 571 octets de données renvoyées dans le corps du message. Le corps contient une mise à jour XMPP concernant un événement d'agent. Le dernier message http-bind a été envoyé par le client Finesse, mais n'a pas reçu de réponse du serveur Finesse. Cela peut être déterminé lorsque vous voyez que le résultat HTTP est nul (-) et que le nombre d'octets dans le corps de la réponse est nul (-1).



Vue rapprochée des données :



6...	200	HTTPS	fin1.ucce.local:...	/http-bind/	1,571		text/xml...	firefo...
6...	202	HTTPS	fin1.ucce.local:...	/finesse/api/User/47...	0	no-ca...	applicatio...	firefo...
6...	200	HTTPS	fin1.ucce.local:...	/desktop/theme/fine...	673		image/gif	firefo...
6...	200	HTTPS	fin1.ucce.local:...	/http-bind/	57		text/xml...	firefo...
6...	200	HTTPS	fin1.ucce.local:...	/finesse/api/SystemI...	232	no-ca...	applicatio...	firefo...
6...	200	HTTPS	fin1.ucce.local:...	/http-bind/	57		text/xml...	firefo...
6...	200	HTTPS	fin1.ucce.local:...	/http-bind/	57		text/xml...	firefo...
6...	200	HTTPS	fin1.ucce.local:...	/finesse/api/SystemI...	232	no-ca...	applicatio...	firefo...
6...	200	HTTPS	fin1.ucce.local:...	/http-bind/	57		text/xml...	firefo...
6...	-	HTTPS	fin1.ucce.local:...	/http-bind/	-1			firefo...
6...	200	HTTPS	fin1.ucce.local:...	/finesse/api/SystemI...	232	no-ca...	applicatio...	firefo...

Corps de réponse pour le message XMPP :


```
<body xmlns='http://jabber.org/protocol/httpbind'><message xmlns="jabber:client" from="pubsub.fin1.ucce.local"
to="47483648@fin1.ucce.local" id="/finesse/api/User/47483648__47483648@fin1.ucce.local__K7hYF"><event
xmlns="http://jabber.org/protocol/pubsub#event"><items node="/finesse/api/User/47483648"><item id="26a3e421-
4752-8a1d-5adbdac74a7717"><notification xmlns="http://jabber.org/protocol/pubsub">&lt;Update&gt;
 &lt;data&gt;
  &lt;user&gt;
   &lt;dialogs&gt;/finesse/api/User/47483648/Dialogs&lt;/dialogs&gt;
   &lt;extension&gt;10005&lt;/extension&gt;
   &lt;firstName&gt;Isaac&lt;/firstName&gt;
   &lt;lastName&gt;Newton&lt;/lastName&gt;
   &lt;loginId&gt;47483648&lt;/loginId&gt;
   &lt;loginName&gt;isaac&lt;/loginName&gt;
   &lt;mediaType&gt;1&lt;/mediaType&gt;
   &lt;pendingState&gt;&lt;/pendingState&gt;
   &lt;roles&gt;
    &lt;role&gt;Agent&lt;/role&gt;
   &lt;/roles&gt;
   &lt;settings&gt;
    &lt;wrapUpOnIncoming&gt;OPTIONAL&lt;/wrapUpOnIncoming&gt;
   &lt;/settings&gt;
   &lt;state&gt;READY&lt;/state&gt;
   &lt;stateChangeTime&gt;2019-01-11T23:56:54.783Z&lt;/stateChangeTime&gt;
   &lt;teamId&gt;5000&lt;/teamId&gt;
   &lt;teamName&gt;Maths&lt;/teamName&gt;
   &lt;uri&gt;/finesse/api/User/47483648&lt;/uri&gt;
  &lt;/user&gt;
 &lt;/data&gt;
 &lt;event&gt;PUT&lt;/event&gt;
 &lt;requestId&gt;07f14a42-6b3c-4855-a4c9-af50ab5e7cc6&lt;/requestId&gt;
 &lt;source&gt;/finesse/api/User/47483648&lt;/source&gt;
&lt;/Update&gt;</notification></item></items></event></message></body>
```


Utiliser Wireshark

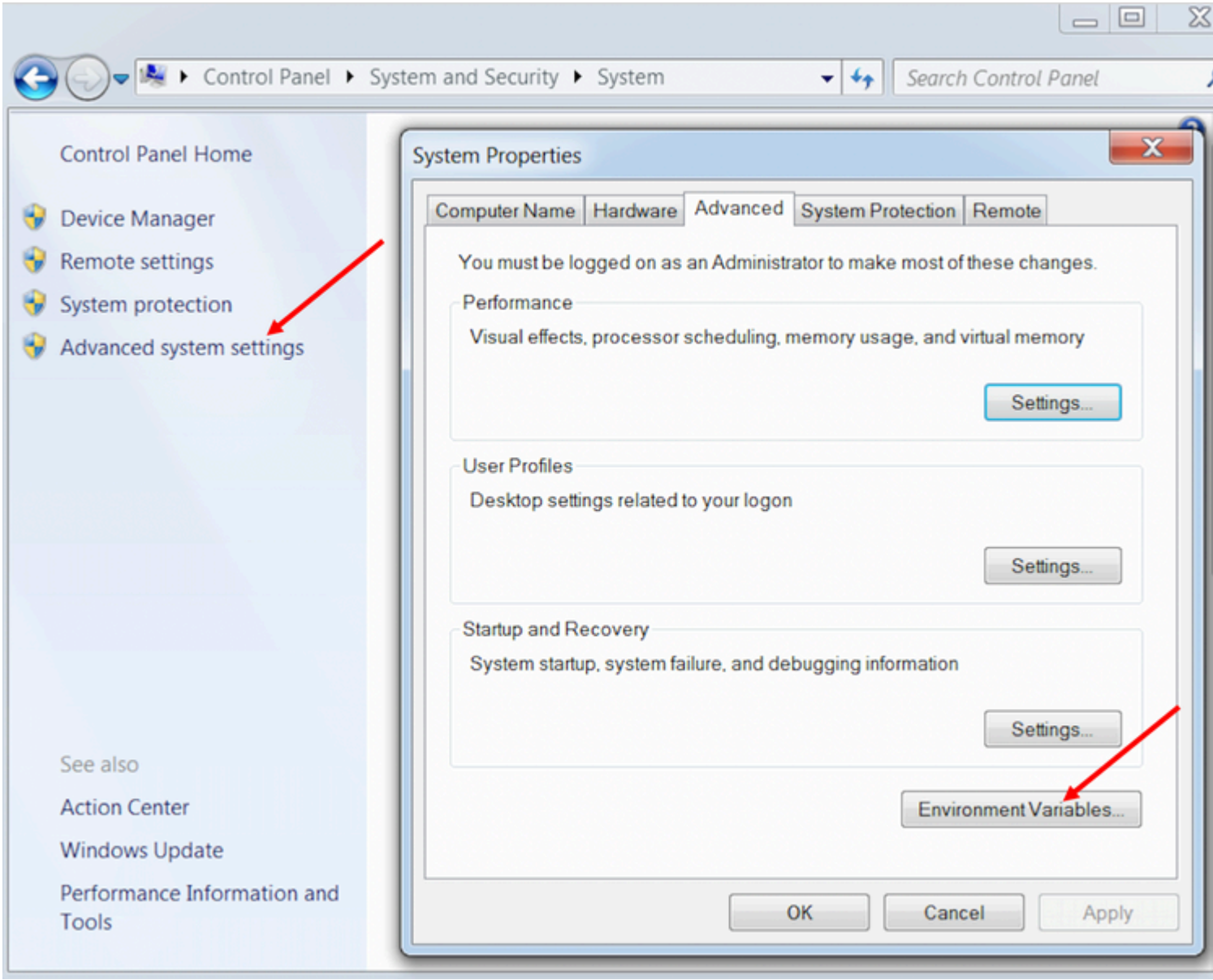
Wireshark est un outil dâ€™analyse de paquets couramment utilisé pour analyser et décoder le trafic HTTPS. Le trafic HTTPS est le trafic HTTP sécurisé sur TLS (Transport Layer Security). TLS assure lâ€™intégrité, lâ€™authentification et la confidentialité entre deux hôtes. Il est couramment utilisé dans les

applications Web, mais il peut être utilisé avec n'importe quel protocole qui utilise TCP comme protocole de couche transport. SSL (Secure Sockets Layer) est l'ancienne version du protocole TLS, qui n'est plus utilisé car il n'est pas sécurisé. Ces noms sont souvent utilisés de manière interchangeable, et le filtre Wireshark utilisé pour le trafic SSL ou TLS est ssl.

Attention : l'exemple de configuration fourni concerne Wireshark 2.6.6 (v2.6.6-0-gdf942cd8) et Mozilla Firefox 64.0.2 (32 bits) sous Windows7 x64 dans un environnement de travaux pratiques. Ces procédures ne peuvent pas être généralisées à toutes les versions de Fiddler, à tous les navigateurs ou à tous les systèmes d'exploitation des ordinateurs. Si votre réseau est actif, assurez-vous de comprendre l'impact potentiel de toute configuration. Pour plus d'informations, consultez la documentation SSL officielle de Wireshark. Wireshark 1.6 ou version ultérieure est requis.

Remarque : cette méthode ne fonctionne que pour Firefox et Chrome. Cette méthode ne fonctionne pas pour Microsoft Edge.

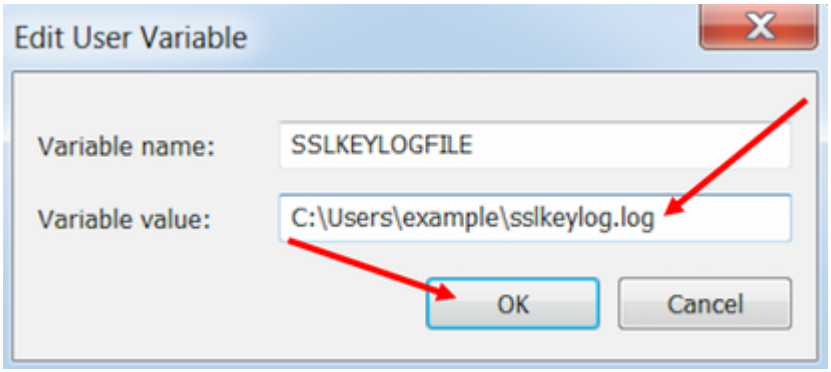
Étape 1. Sur le PC Windows de l'agent, accédez à **Panneau de configuration** > **Système et sécurité** > **Système** > **Paramètres système avancés Variables environnementales...**



Étape 2. Accédez aux **variables utilisateur pour l'utilisateur <username> > Nouveau...**

Créez une variable nommée **SSLKEYLOGFILE.**

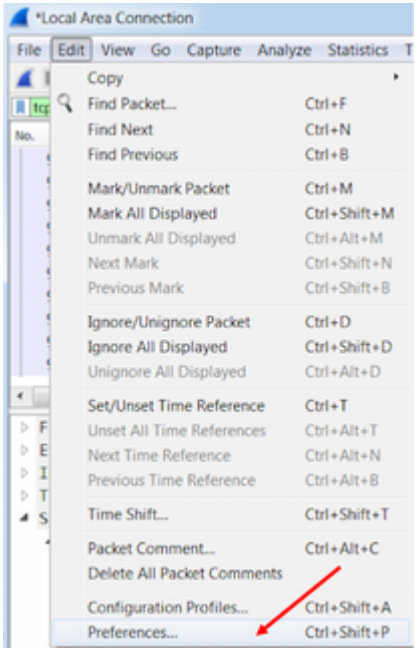
Créez un fichier pour stocker le secret de prémaître SSL dans un répertoire privé : SSLKEYLOGFILE=</path/to/private/directory/with/logfile>



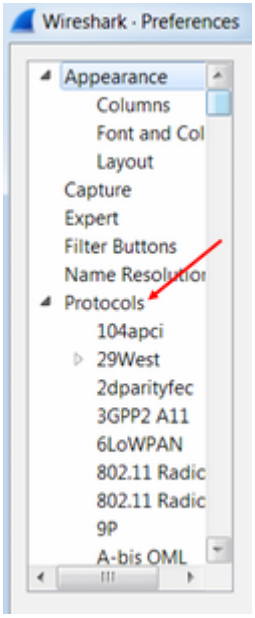
> **Remarque** : créez une variable système au lieu d'une variable utilisateur et/ou stockez le fichier dans un répertoire non privé, mais tous les utilisateurs du système peuvent accéder au secret prémaître, qui est moins sécurisé.

Étape 3. Si Firefox ou Chrome sont ouverts, fermez les applications. Une fois rouverts, ils peuvent commencer à écrire dans le fichier SSLKEYLOGFILE.

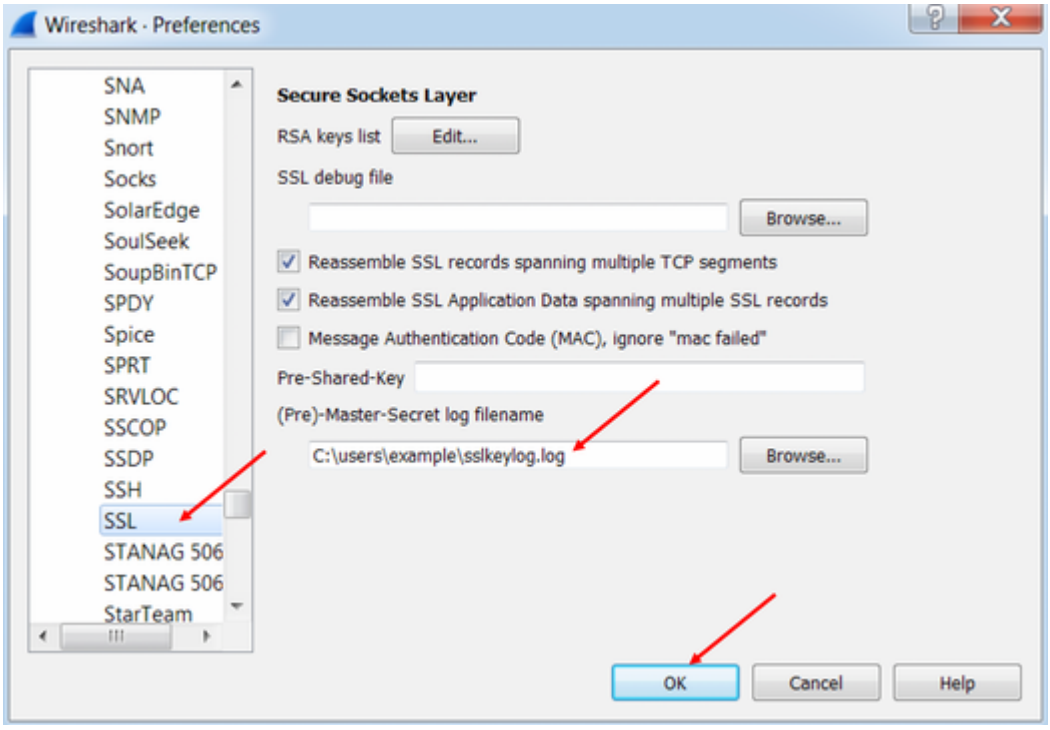
Étape 4. Sur Wireshark, accédez à **Edit > Preferences...**



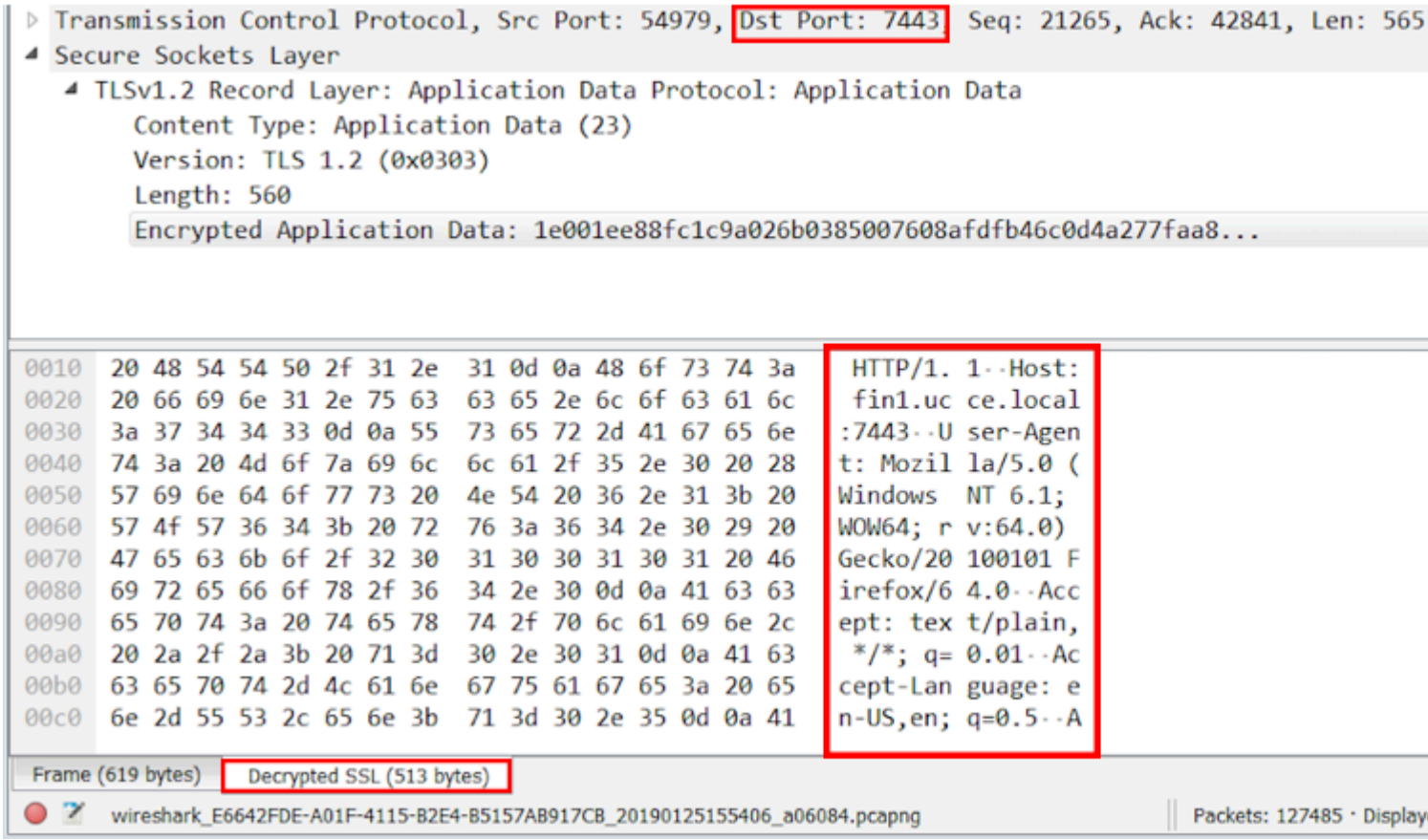
Accédez à **Protocoles > SSL**.



Étape 5. Entrez l'emplacement du nom de fichier journal secret prémaître configuré à l'étape 2.



Étape 6. Utilisez le filtre Wireshark **tcp.port==7443 && ssl**, la communication HTTP sécurisée entre le client Finesse et le serveur Finesse (Service de notification) est vue déchiffrée.



Défauts associés

- ID de bogue Cisco CSCva7280 - Tomcat Finesse et crash d'Openfire pour caractères XML non valides
- ID de bogue Cisco CSCva72325 - UCCX : Finesse Tomcat et Openfire Crash pour caractères XML non valides

Informations connexes

- Spécifications XMPP
- XEP-0124 : BOSH
- XEP-0060 : Publier-S'abonner
- Console Web Firefox
- Console Web Microsoft Edge
- Console Web Chrome
- Windows PowerShell
- Analyseur de performances Windows
- Dépannage des suppressions dans la file d'attente d'entrée et de sortie
- Gestionnaire des tâches Windows
- Terminal Mac
- Moniteur d'activité Mac
- Téléchargement de Fiddler
- Configuration de violon
- Téléchargement Wireshark
- Décodage SSL Wireshark
- Assistance et documentation techniques - Cisco Systems

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.