

Comment la connexion de bêtises fonctionne pour la présence d'Agent Desktop de finesse

Contenu

[Introduction](#)

[Quelles sont des BÊTISES ?](#)

[Problème courant dû au débranchement de BÊTISES](#)

[Analyse de log](#)

[Logs de debug](#)

[Logs de l'information](#)

[Logs de Webservices](#)

[Raisons communes pour le débranchement de BÊTISES](#)

[Actions de suggestion](#)

Introduction

Ce document décrit l'architecture derrière des connexions de finesse qui utilise des Bidirectionnel-flots au-dessus de HTTP synchrone (BÊTISES) et comment le problème lié à ceci peut être diagnostiqué.

Quelles sont des BÊTISES ?

La Messagerie et la présence extensibles Protocol (XMPP) est état de protocole dans un client-server model. Si une application doit fonctionner avec XMPP, les plusieurs questions surgissent. Les navigateurs ne prennent en charge pas XMPP à la façon des indigènes, ainsi tout le trafic XMPP doit être traité par un programme qui fonctionne à l'intérieur du navigateur.

Le premier problème est que le HTTP est protocole sans état. Ceci signifie que chaque demande de HTTP n'est liée à aucune autre demande. Cependant, ce problème peut être abordé par applicable signifie--par exemple à l'aide des Témoins/données de courrier.

Le deuxième problème est le comportement unidirectionnel du HTTP. Seulement le client envoie des demandes, et le serveur peut seulement répondre. L'incapacité du serveur de pousser des données le rend artificiel d'implémenter XMPP au-dessus de HTTP.

Ce problème est éliminé si le client peut faire des demandes directes de TCP (de ce fait éliminant le besoin du HTTP). Cependant, si vous voulez aborder le problème dans le domaine de HTTP (par exemple parce que le Javascript peut envoyer des demandes de HTTP), il y a deux solutions possibles. Chacun des deux exigent une passerelle entre le HTTP et le XMPP. Les solutions vote (des demandes de HTTP répétées demandant de nouvelles données) et de longues interrogations, également connues sous le nom de BÊTISES.

Le but d'utiliser des BÊTISES est de dissimuler le fait que le serveur ne doit pas répondre car bientôt il y a une demande. La réponse est retardée (un long balayage) jusqu'à ce que le serveur ait des données pour le client, et alors il est envoyé comme réponse. Dès que le client l'obtiendra il fait une nouvelle demande (même si il n'a rien à envoyer) et ainsi de suite.

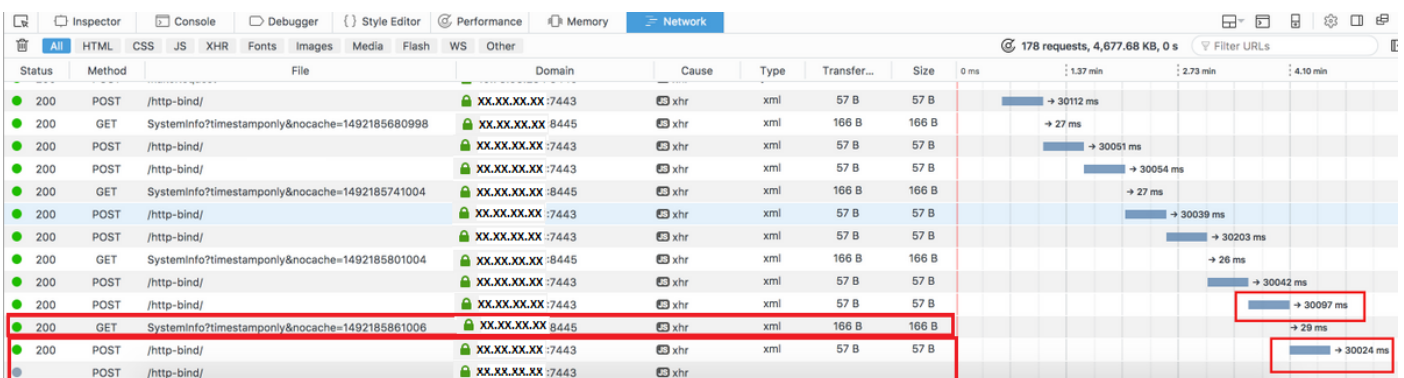
Les BÊTISES sont tout à fait efficaces du point de vue de chargement du serveur et en ce qui concerne le trafic.

Le client de bureau de finesse (application Web) établit une connexion éventée de BÊTISES toutes les 30 secondes. Après 30 secondes, s'il n'y a aucune mise à jour du service de Notificaiton de finesse, le service de notification envoie une réponse de HTTP avec un OK 200 et (presque) un corps vide de réponse. Si le service de notification a une mise à jour sur la présence d'un agent ou un événement de dialogue (appel), par exemple, les données sont envoyées immédiatement au client web de finesse.

Pour récapituler :

1. Le client web de finesse aura toujours une connexion HTTP éventée (HTTP-grippage) installée au serveur de finesse par l'intermédiaire du port TCP 7443. Ceci est connu comme long balayage de BÊTISES.
2. Le service de notification de finesse est un service de présence qui signale des mises à jour concernant l'état d'un agent, d'un appel, etc.
3. Si le service de Notiication a une mise à jour, il répondra à la demande de HTTP-grippage avec la mise à jour d'état comme message XML dans le corps de réponse de HTTP.
4. S'il n'y a aucune mise à jour 30 d'état des secondes après réception de la demande de HTTP-grippage, le service de Notificaiton ne répond sans aucune mise à jour d'état pour permettre au web client de finesse pour envoyer une autre demande de HTTP-grippage. Ceci sert de manière au service de notification de savoir que le client web de finesse peut encore se connecter au service de notification et que l'agent n'a pas fermé leur navigateur ou a mis leur ordinateur pour dormir, etc.

En outre, le client web de finesse envoie une demande de SystemInfo qui vérifie l'état de Tomcat de finesse chaque minute, suivant les indications de l'image :



Suivant les indications de l'image, le service de notification (Openfire) mettent au point le log affiche le HTTP liant avec l'appareil de bureau avec l'adresse IP et le port du PC d'agent.

```

XXX.XXX.XXX.XX:1:34:21 [Session=1, SSL_NULL_WITH_NULL_NULL] received @ sent @
2017.04.14 21:34:21 REQUEST /http-bind/ on org.eclipse.jetty.server.nio.SelectChannelConnector$SelectChannelHttpConnection@2d4526@XXX.XXX.XXX.XX:7443<->XXX.XXX.XXX.XX:49805
2017.04.14 21:34:21 scope null[/http-bind/ @ o.e.j.s.ServletContextHandler{/http-bind,null}]
2017.04.14 21:34:21 context=/http-bind[/ @ o.e.j.s.ServletContextHandler{/http-bind,null}]
2017.04.14 21:34:21 sessionManager=org.eclipse.jetty.server.session.HashSessionManager@176fe4#STARTED
2017.04.14 21:34:21 session=null
2017.04.14 21:34:21 session=null
2017.04.14 21:34:21 servlet /http-bind[/ -> org.jivesoftware.openfire.http.HttpBindServlet-1643193
2017.04.14 21:34:21 chain=null
2017.04.14 21:34:21 HTTPBindLog: HTTP RECVC(3445afbe): <body sid="3445afbe" rid="164853266"/>
2017.04.14 21:34:21 consumeResponse: org.jivesoftware.openfire.http.HttpSession@d47653 - status: 3 address: 1001003@XXX.XXX.XXX.XX.cisco.com/desktop id: 3445afbe presence:
<presence from="1001003@XXX.XXX.XXX.XX.cisco.com/desktop">
<x xmlns="http://jabber.org/protocol/caps" hash="sha-1" node="http://jabber.cisco.com/caxl" ver="VNC6fNwvCxe6FJfJdJpLryVJRwM"/>
</presence> rid: 164853266
2017.04.14 21:34:21 suspended org.eclipse.jetty.server.nio.SelectChannelConnector$SelectChannelHttpConnection@2d4526@XXX.XXX.XXX.XX:7443<->XXX.XXX.XXX.XX:49805
2017.04.14 21:34:24 Launching thread for /127.0.0.1:44667
2017.04.14 21:34:24 Launching thread for /127.0.0.1:44656
    
```

Suivant les indications de l'image, le **dernier ms de l'active 0** prouve que la session est encore en activité.

```
2017.04.14 21:34:26 Exiting since queue is empty for /127.0.0.1:44678
2017.04.14 21:34:26 Exiting since queue is empty for /127.0.0.1:44678
2017.04.14 21:34:26 Session (id=3445afbe) was last active 0 ms ago: 1001003@XXXXXXXX.XXXXXXXXXX.cisco.com/desktop
2017.04.14 21:34:26 time=1492185866851,JID=1001003@XXXXXXXX.XXXXXXXXXX.cisco.com/desktop,msgs_sent=4,msgs_queue=0,msgs_drop=0,bytes_sent=3748
2017.04.14 21:34:26 time=1492185866851,JID=1001003@XXXXXXXX.XXXXXXXXXX.cisco.com/desktop,msgs_sent=4,msgs_queue=0,msgs_drop=0,bytes_sent=3748
2017.04.14 21:34:26 Launching thread for /127.0.0.1:44678
2017.04.14 21:34:26 Exiting since queue is empty for /127.0.0.1:44678
```

Le service de notification d'enable mettent au point des logs d'Unified Contact Center Express (UCCX), suivant les indications de l'image :

```
[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 will be disabled automatically if Cisco Unified CCX Notification Service is restarted.
admin:█
```

Le service de notification d'enable mettent au point des logs d'Unified Contact Center Enterprise (UCCE) (finesse autonome), suivant les indications de l'image :

```
[admin:utils finesse notification logging enable

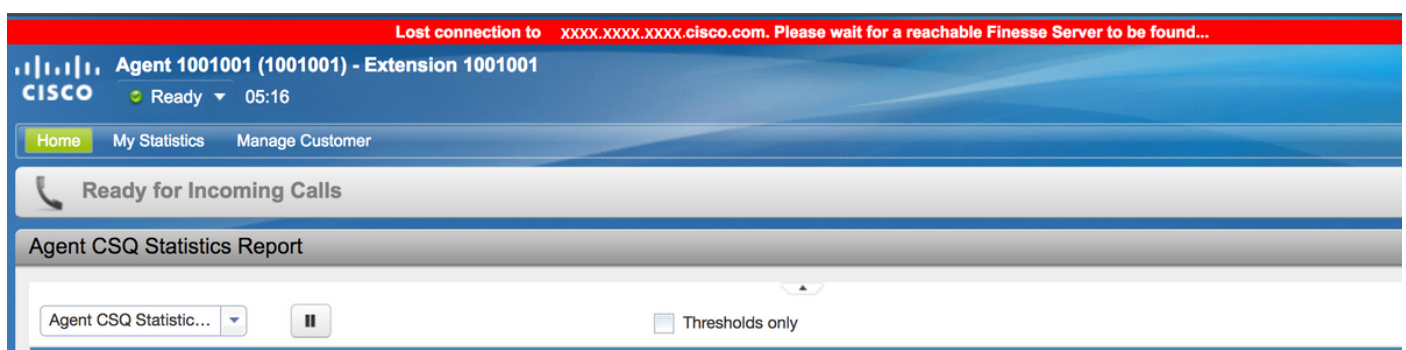
Checking that the Cisco Finesse Notification Service is started...
The Cisco Finesse Notification Service is started.

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 will be disabled automatically if you restart the Cisco Finesse Notification Service.
```

Problème courant dû au débranchement de BÊTISES



Pour UCCX, le temporisateur de déconnexion est de 60 secondes après que débranchement de navigateur. L'agent peut être dans l'état prêt ou non prêt pour que la déconnexion se produise.

Pour UCCE, la finesse prend à 120 secondes pour la détecter quand un agent ferme le navigateur ou le navigateur tombe en panne et attente de finesse 60 secondes avant d'envoyer une demande obligatoire de déconnexion au serveur CTI. Dans ces conditions, la finesse peut prendre à 180 secondes pour se déconnecter l'agent.

Pour plus d'informations sur le comportement de bureau de finesse UCCE, référez-vous à la section *de bureau de comportement* du chapitre de *mécanismes de Basculement de Cisco*

Finesse dans le [guide d'administration de Cisco Finesse](#).

Analyse de log

Des logs du service de finesse et de Notification peuvent être collectés par l'intermédiaire de RTMT ou par l'intermédiaire du CLI :

le fichier obtiennent l'activelog /desktop se reproduit compressé

Logs de debug

Ces logs sont dans le répertoire de /desktop/logs/openfire et sont nommés debug.log.

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

2017.06.17 00:17:04 Unable to route packet. No session is available so store offline. <message
from="pubsub.xxxxx.xxxx.xxx.cisco.com" to="1001003@xxxxxx.xxxx.xxx.cisco.com.cisco.com"
id="/finesse/api/User/1001003__1001003@xxxxxx.xxxx.xxx.cisco.com__o5Aqb"><event
xmlns="http://jabber.org/protocol/pubsub#event"><items node="/finesse/api/User/1001003"><item
id="0d78a283-466d-4477-a07e-6e33a856f388"><notification
xmlns="http://jabber.org/protocol/pubsub">&lt;Update&gt;
```

Logs de l'information

Ces logs sont dans le répertoire de /desktop/logs/openfire et sont nommés info.log.

```
2017.06.17 00:16:04 Closing idle session (id=f382a015): 1001003@xxxxxx.xxxx.xxx.
cisco.com/desktop after being inactive for more than threshold value of 60
2017.06.17 00:16:04 A session is being closed for 1001003@xxxxxx.xxxx.xxx.cisco.com/desktop
Openfire en clôturant la session de veille indique que déconnexion d'agent déclenche 1 minute
et ceci apparaît comme code de raison 255 dans les états
```

Logs de Webservices

Ces logs sont dans le répertoire de /desktop/logs/webservices et sont nommés Desktop-webservices.YYYY-MM-DDTHH-MM-SS.sss.log.

```
0000001043: XX.XX.XX.XXX: Jun 17 2017 00:16:04.630 +0530: %CCBU_Smack Listener Processor (1)-6-
PRESENCE_NOTIFICATION_RECIEVED: %[FROM
JID=1001003@xxxxxx.xxxx.xxx.cisco.com/desktop][PRESENCE_TYPE=unavailable]:Finesse received a
presence notifcation
0000000417: XX.XX.XX.XXX: Jun 17 2017 00:16:04.631 +0530: %CCBU_Smack Listener Processor (1)-6-
UNSUBSCRIBE_REQUEST_SUCCESS:
%[NodeId=/finesse/api/User/1001003/ClientLog][user_id=1001003@xxxxxx.xxxx.xxx.cisco.com]:
Sucessfully unsubscribed from a node on the XMPP server
```

```
0000001044: XX.XX.XX.XXX: Jun 17 2017 00:16:04.631 +0530: %CCBU_Smack Listener Processor (1)-6-AGENT_PRESENCE_MONITOR: %[message_string=Adding agent 1001003 into the expiry hash.]:
0000001051: XX.XX.XX.XXX: Jun 17 2017 00:16:35.384 +0530: %CCBU_pool-8-thread-1-6-AGENT_PRESENCE_MONITOR: %[message_string=[Expired] Removed agent from cache 1001003]:
0000001060: XX.XX.XX.XXX:: Jun 17 2017 00:17:04.632 +0530: %CCBU_CoreImpl-worker12-6-PRESENCE DRIVEN LOGOUT: %[agent_id=1001003]: Performing CTI Logout on basis of the agents unavailable presence
0000001061: XX.XX.XX.XXX:: Jun 17 2017 00:17:04.633 +0530: %CCBU_CoreImpl-worker12-6-MESSAGE_TO_CTI_SERVER: %[cti_message=Invoke id :39 , agentstate : 1, workmode : 0, reason code: 255, forceflag :1, agentcapacity: 1, agenttext: 1001003, agentid: 1001003, supervisorid: null, ssoFlag=false][cti_message_name=SetAgentStateReq]: Message going to the backend cti server
0000001066: XX.XX.XX.XXX:: Jun 17 2017 00:17:04.643 +0530: %CCBU_CTI_MESSAGE_EVENT_EXECUTOR-0-6-DECODED_MESSAGE_FROM_CTI_SERVER: %[cti_message=CTI AgentStateEvent [skillGroupState=1 (LOGOUT), stateDuration=0, skillGroupNumber=-1, skillGroupPriority=0, agentState=1 (LOGOUT), eventReasonCode=255, numFltSkillGroups=0, CTIClientSignature=null, agentID=1001003, agentExtension=1001003, agentInstrument=null, agentID_Long=1001003, duration=null, nextAgentState=null, fltSkillGroupNumberList=[], fltSkillGroupIDList=[], fltSkillGroupPriorityList=[], fltSkillGroupStateList=[], MRDId=1, agentMode=0]CTI MessageBean [invokeID=null, cti_sequence_id=105, msgID=30, timeTracker={"id": "AgentStateEvent", "CTI_MSG_RECEIVED":1497638824642, "CTI_MSG_DISPATCH":1497638824643}, msgName=AgentStateEvent, deploymentType=CCX]][cti_response_time=1][dispatch_phase=DnD-CHECKPOINT-3B]: Decoded Message to Finesse from backend cti server
```

Raisons communes pour le débranchement de BÊTISES

- Problème de réseau
- Navigateur et/ou version non vérifiés.
- Le navigateur a collé la condition due à pour contenter de l'autres onglet/fenêtre.
- Ordinateur mis pour dormir.
- Question CPU de haute ou de mémoire élevée sur le serveur.
- instruments de tiers faisant l'activité de malice à l'arrière-plan.
- Connexion perdante d'engine CTI Server/CCX avec le serveur de finesse.
- Question de NTP sur le serveur ou le client.

Actions de suggestion

- Utilisez le navigateur/version et les configurations pris en charge selon les guides.
- Testez le comportement tous les instruments de tiers étant coupés.
- Comprenez le comportement et le flux des tâches d'agent.
- Cisco Finesse Tomcat de reprise et service de notification.
- Exécutez les **utils diagnostiquent le test** sur le serveur, et valident la consultation en avant et inverse de DN du serveur et du client chacun des deux.

- **État de ntp d'utilis de** contrôle sur le serveur.
- Dans les logs de client, la dérive et la latence de réseau de contrôle. Des logs de client peuvent être vus des logs de console Web de navigateur ou en appuyant sur **l'état d'erreur d'envoi à la** page de finesse et en collectant les logs de finesse. Les logs se trouvent dans **/desktop/logs/clientlogs**.

Note: Les valeurs de temporisateur pourraient changer à l'avenir selon la condition requise de produit.