

# Login-Overtrek van FineReader Agent met het gebruik van Logs

## Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[Start het bureaublad](#)

[Credentials agent voor aanmelding](#)

[Systeeminfo](#)

[API AANVRAAG](#)

[De BOSH-verbinding opzetten](#)

[Agent-aanmelding](#)

[Aanmelden uitvoeren](#)

[Uitlogcodes, reden, telefoon, boek](#)

## Inleiding

Dit document beschrijft het proces dat gepaard gaat met de inlogging van een agent via het Finse systeem met de logbestanden. Het is belangrijk om de berichtstroom tussen de verschillende componenten Finesse, de Server van de Integratie van de Computer Telephony (CTI), en de desktop te begrijpen zodat u met succes problemen kunt oplossen.

## Voorwaarden

### Vereisten

Cisco raadt u aan kennis te hebben van Cisco Finesse en de opdrachtmelding van Voice Operating System (VOS) CLI.

### Gebruikte componenten

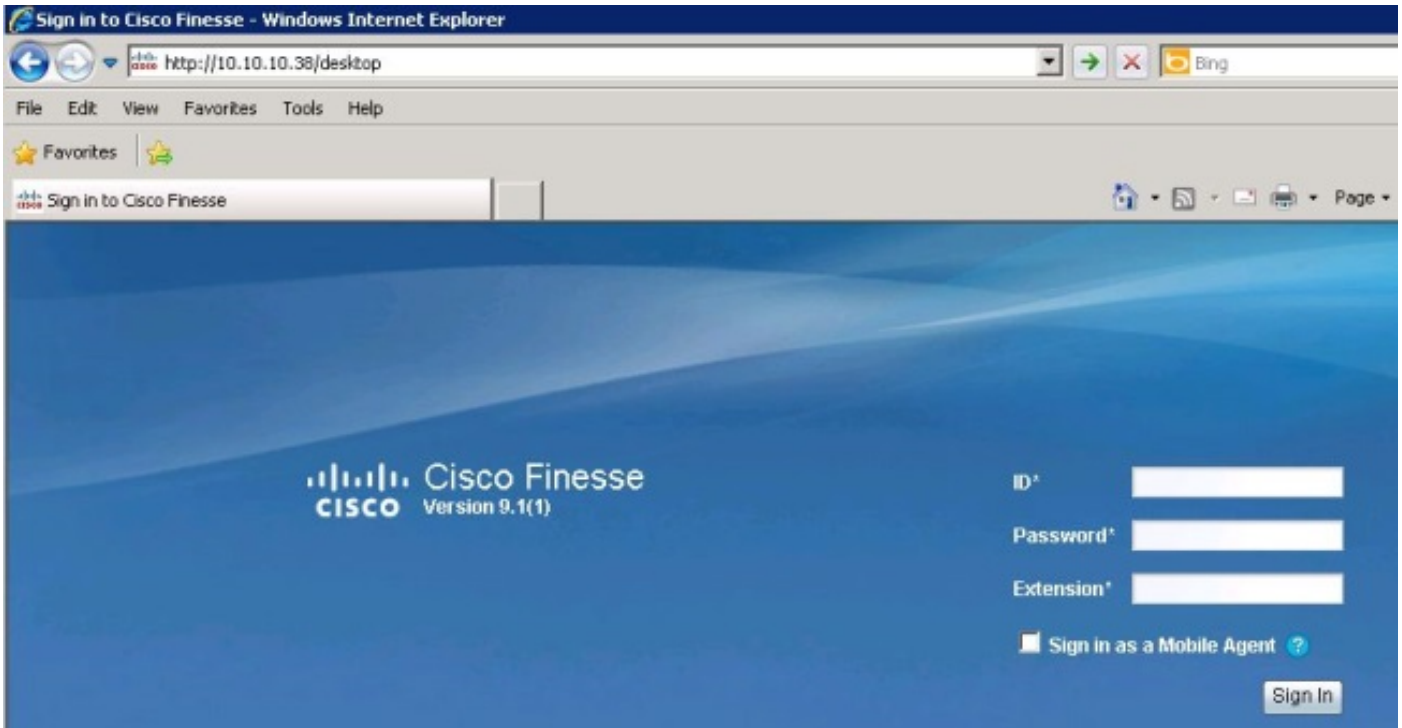
De informatie in dit document is gebaseerd op Cisco Finse versie 9.1(1).

De informatie in dit document is gebaseerd op de apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als uw netwerk live is, moet u de potentiële impact van

elke opdracht begrijpen.

## Start het bureaublad

Om het bureaublad te lanceren, kopieert u deze URL in de webbrowser: **<your finesse server>/desktop**. In Finesse versie 9.1 wordt HTTP of HTTPS ondersteund.



Finesse gebruikt Tomcat als webserver. Wanneer u uw webbrowser start, wordt het verzoek aan Finesse gedaan om het bureaublad aan u voor te leggen. De opdracht Cisco Tomcat **localhose\_access\_log** toont het verzoek om het bureaublad te laden.

```
10.10.10.211 10.10.10.211 - - 80 GET / HTTP/1.1 302 - 141
10.10.10.211 10.10.10.211 - - 80 GET /desktop/container/ HTTP/1.1 200 4541 185
10.10.10.211 10.10.10.211 - - 80 GET /desktop/theme/finesse/css/base.css
HTTP/1.1 200 3093 7
10.10.10.211 10.10.10.211 - - 80 GET /desktop/theme/finesse/css/login.css
HTTP/1.1 200 2185 0
10.10.10.211 10.10.10.211 - - 80 GET /desktop/js/Logon.js HTTP/1.1 200 1745 1
10.10.10.211 10.10.10.211 - - 80 GET /desktop/js/utilities/Cookies.js HTTP/1.1
200 2390 0
10.10.10.211 10.10.10.211 - - 80 GET /desktop/thirdparty/jquery/js/jquery.tools.
min.js HTTP/1.1 200 15699 1
10.10.10.211 10.10.10.211 - - 80 GET /desktop/thirdparty/jquery/js/jquery-1.5.
min.js HTTP/1.1 200 84523 7
10.10.10.211 10.10.10.211 - - 80 GET /desktop/theme/finesse/images/sprite_
buttons.png HTTP/1.1 200 3297 0
10.10.10.211 10.10.10.211 - - 80 GET /desktop/theme/finesse/images/help.png
HTTP/1.1 200 830 0
10.10.10.211 10.10.10.211 - - 80 GET /desktop/theme/finesse/images/cisco_logo.
png HTTP/1.1 200 760 0 200 2205 1
10.10.10.211 10.10.10.211 - - 80 GET /desktop/theme/finesse/images/bg.jpg
HTTP/ 1.1 200 32222 4
```

# Credentials agent voor aanmelding

Nu het bureaublad is voorgesteld, gaat u uw inloggehoofsbriefen in. Voordat Finesse de inlogaanvraag naar de CTI server kan verzenden, moet de client een Bidirectional-streams over Synchronous HTTP (BOSH)-verbinding instellen. Om de BOSH Connection te kunnen opzetten, vraagt de client eerst systeem informatie van de Finse Server aan.

## Systeminfo

Het bureaublad van de klant heeft een verzoek gedaan van de Representational State Transfer (REST) Application Programming Interface (API) aan deze URL: **/finesse/api/SystemInfo**. Neem nota van de **nocache=**. Deze unieke ID wordt gebruikt om dit verzoek via het systeem te kunnen traceren. **Teruggezonden met status=200** geeft aan dat het verzoek met succes is ontvangen.

```
Container : [ClientServices] SystemInfo: requestId='undefined', Making REST
request: method=GET, url='/finesse/api/SystemInfo?nocache=1366756802163' 18:40:03:
Container : [ClientServices] SystemInfo: requestId='undefined', Returned
with status=200
```

Als u geen clientlogs hebt maar u het verzoek moet overtrekken, kunt u de Tomcat **localhost\_access\_log** doorzoeken om te bepalen wanneer de REST API-aanvraag is ingediend en de unieke identificator te vinden.

```
127.0.0.1 127.0.0.1 - - 80 GET /finesse/api/SystemInfo ?nocache=1366756802163
HTTP/1.1 200 336 120 10.10.10.211 10.10.10.211 2001 - 80 GET /gadgets/makeRequest
?refresh=3600&url=http%3A%2F%2Flocalhost%2Ffinesse%2Fapi%2FSystemInfo%3Fnocache%
3D1366756802163&httpMethod=GET&headers=Authorization%3DBasic%2520MjAwMToyMDAx%
26locale%3Den_US&postData=&authz=&st=&contentType=TEXT&numEntries=3&getSummaries
=false&signOwner=true&signViewer=true&gadget=undefined&container=default&
bypassSpecCache=&getFullHeaders=false HTTP/1.1 200 659 596
```

## API\_AANVRAAG

Tomcat stuurt dit API-verzoek naar de Finse REST API-webtoepassingsopslag (WAR). Om de Finesse REST API-logbestanden te vinden, zoekt u het Finse webservice-logbestand via timestamp of de nocache-ID om de API\_REQUEST te vinden. Dit logbestand toont **REQUEST\_START**, **REQUEST\_URL**, **REQUEST\_END**, **EIST\_START** en the **Verleapsed\_time** het systeem heeft de aanvraag voltooid.

```
%CCBU_http-8080-7-6-REQUEST_START: %[method_name=GET][parameter_name=
{ nocache=1366756802163, }][resource_name=/SystemInfo][usr]: Request start
%CCBU_http-8080-7-6-API_REQUEST: %[REQUEST_URL=SystemInfo][agent_id=][request_
identifier=][request_method=systemInfo.GET][request_parameters=]: Request from
client to webservice api
%CCBU_http-8080-7-6-REGISTER_API_STATS_OBJECT: %[resource_name=com.cisco.ccbu:
category=WebAppStats,component0=SystemInfo-GET]: Registered new api stats object
for new request type. %CCBU_http-8080-7-6-REQUEST_END: %[elapsed_time=98]: Request complete
```

De inhoud die door het REST API-verzoek aan de client wordt teruggegeven om systeem informatie op te halen, wordt hier getoond. Deze informatie bevindt zich in de client (agent) logboeken.

```
content='<SystemInfo>
<primaryNode>
<host>UCCEFINESSSE91.vmlod.cvp</host>
</primaryNode>
<secondaryNode>
<host>UCCEFINESSSE138.vmlod.cvp</host>
</secondaryNode>
<status>IN_SERVICE</status>
<xmppDomain>UCCEFINESSSE138.vmlod.cvp</xmppDomain>
<xmppPubSubDomain>pubsub.UCCEFINESSSE138.vmlod.cvp</xmppPubSubDomain>
</SystemInfo>'
```

## De BOSH-verbinding opzetten

Het SystemInfo geeft de primaire en secundaire Finesse servers, de status van Finesse als **IN\_SERVICE**, het **xmppDomain** en het **EXMPPPubSub** weer. De cliënt heeft nu genoeg informatie om een BOSH verbinding te maken.

```
18:40:03: Container : PageServices.init().onLoad: System info status: IN_SERVICE
18:40:03: Container : PageServices.init(): Establishing BOSH connection...
18:40:03: Container : PageServices.init(): Starting timeout and poller...
18:40:04: Container : [ClientServices] MasterPublisher._connInfoHandler()
- Connection status: connecting
18:40:04: Container : [ClientServices] MasterPublisher._connInfoHandler()
- Connection status: disconnected
18:40:04: Container : PageServices._onDisconnect(): retryCount=0, retrying...
18:40:04: Container : [ClientServices] MasterPublisher._connInfoHandler()
- Connection status: connecting
18:40:05: Container : [ClientServices] MasterPublisher._connInfoHandler()
- Connection status: connected
18:40:05: Container : PageServices.onLoad(): BOSH established!
```

De client is geabonneerd op Finse Object (knooppunt) **/finesse/api/User/2001** nadat de BOSH-verbinding is opgezet.

Wanneer de BOSH verbinding van de cliënt wordt gevestigd, ontvangt het web services logbestand een **PRESENCE\_NOTIFICATION** bericht van de cliënt. Deze **PRESENCE\_TYPE** geeft alleen aan dat de client beschikbaar is om **XMPP-gebeurtenissen** te ontvangen en heeft niets te maken met de beschikbaarheid van de agent in Unified Contact Center Enterprise (UCCE). Onthoud dat de agent nog niet is ondertekend.

Opmerking: U ziet de **PRESENCE\_TYPE** berichten alleen wanneer een client een BOSH-verbinding instelt of wanneer de BOSH-verbinding van een client is losgekoppeld. Wanneer de BOSH-verbinding van de client is losgekoppeld, wordt **PRESENCE\_TYPE** weergegeven als niet beschikbaar.

Dit is de melding in het weblog van internetdiensten:

```
%CCBU_Smack Listener Processor (1)-6-PRESENCE_NOTIFICATION_RECIEVED:
%[FROM JID=2001@uccefinessse138.vmlod.cvp/desktop]
[PRESENCE_TYPE=available]: Finesse received a presence notifcation
```

## Agent-aanmelding

Nu de cliënt de BOSH verbinding heeft gevestigd, begint het intekenproces. De client doet een ander REST API-verzoek om actuele gebruikersinformatie te verkrijgen. Om dit verzoek te kunnen indienen, navigeer dan naar deze URL: **/finesse/api/User/2001** en voer methode=**GET** in.

Omdat dit een ander API-verzoek is, is de **nocache-ID** anders. Om dit verzoek te kunnen volgen, moet je deze nieuwe ID gebruiken.

```
Container : PageServices.onLoad(): BOSH established! Commencing sign-in process
Container : [ClientServices] User: requestId='undefined', Making REST request:
method=GET, url='/finesse/api/User/2001?nocache=1366756805180
'18:40:05: Container : [ClientServices] User: requestId='undefined',
Returned with status=200,
```

U kunt dit verzoek indien nodig vinden in het Tomcat **localhost\_access\_log**. Op deze manier vind je het in het weblog:

```
%CCBU_http-8080-7-6-REQUEST_START: %[method_name=GET][parameter_name={ nocache=
[1366756805180], }][resource_name=/User/2001][usr=2001]: Request start
%CCBU_http-8080-7-6-API_REQUEST: %[REQUEST_URL=User/2001][agent_id=2001]
[request_identifier=null][request_method=user.GET][request_parameters=2001]:
Request from client to webservice api
```

Hier is het verzoek in het logbestand van de Meldingsservices. Neem nota van **HTTP/1.1 200 ok**.

Opmerking: Het Cisco-aanmeldingslogboek is alleen ter informatie. Als u Cisco Finesse-melding toestaat, heeft dit gevolgen voor de prestaties.

```
>> "GET /finesse/api/User/2001 HTTP/1.1[\r][\n]"
Adding Host request header
>>"Authorization: Basic MjAwMToyMDAx[\r][\n]"
>>"User-Agent: Jakarta Commons-HttpClient/3.1[\r][\n]"
>>"Host: localhost:8080[\r][\n]"
>>"[\r][\n]"
<<"HTTP/1.1 200 OK[\r][\n]"
<<"HTTP/1.1 200 OK[\r][\n]"
<<"Pragma: No-cache[\r][\n]"
<<"Cache-Control: no-cache[\r][\n]"
```

Nu de Aanmeldingsdienst het verzoek heeft ingediend, publiceert deze de informatie voor deze gebruiker. Dit is de POST uit het Meldingssservicelampje dat naar de client gaat:

```
Cookie accepted: "$Version=0; JSESSIONIDSSO=C11F62C59D0D0438CDEDEEB0DB12AA0B;
$Path=/"
Cookie accepted: "$Version=0; JSESSIONID=25FE81BD7DB73280A07B4CA4138E7680;
$Path=/finesse"
Buffering response body
<<"<User>[\n]"
<<" <dialogs>/finesse/api/User/2001/Dialogs</dialogs>[\n]"
<<" <extension></extension>[\n]"
<<" <firstName>Mickey</firstName>[\n]"
<<" <lastName>Mouse</lastName>[\n]"
<<" <loginId>2001</loginId>[\n]"
<<" <loginName>mmouse</loginName>[\n]"
<<" <roles>[\n]"
<<" <role>Agent</role>[\n]"
<<" </roles>[\n]"
<<" <state>LOGOUT</state>[\n]"
```

```
<<" <stateChangeTime></stateChangeTime>[\n] "  
<<" <teamId>5000</teamId>[\n] "  
<<" <teamName>Minnies_Team</teamName>[\n] "  
<<" <uri>/finesse/api/User/2001</uri>[\n] "  
<<"</User>"
```

Deze **XMPP Event**, dat **agent 2001** in dit voorbeeld is, wordt naar alle abonnementscliënten verzonden. De JavaScript op de client ontvangt de **XMPP Event** en de gebeurtenis wordt naar het Gadget binnen de client verzonden. Hier zijn de clientlogs die de inhoud van de respons laten zien:

```
Commencing sign-in process18:40:05: Container : [ClientServices] User: requestId=  
'undefined', Maul= '/finesse/api/User/2001?nocache=1366756805180'18:40:05:  
Container : [ClientServices] User: requestId='undefined', Returned with status=200,  
content='<User> king REST request: method=GET,  
<dialogs>/finesse/api/User/2001/Dialogs</dialogs>  
<extension></extension>  
<firstName>Mickey</firstName>  
<lastName>Mouse</lastName>  
<loginId>2001</loginId>  
<loginName>mmouse</loginName>  
<roles>  
<role>Agent</role>  
</roles>  
<state>LOGOUT</state>  
<stateChangeTime></stateChangeTime>  
<teamId>5000</teamId>  
<teamName>Minnies_Team</teamName>  
<uri>/finesse/api/User/2001</uri>  
</User>
```

## Aanmelden uitvoeren

Nu is de client klaar om inloggen uit te voeren. Opmerking over de **aanvraag-ID**. De aanvraag-ID wordt in de tekst van het verzoek verzonden. U gebruikt deze ApplicationID om de inlogaanvraag naar de **REST API > CTI > REST API > Meldingsservice > Reactie** op de client te volgen. Dit verzoek is een PUT, wat betekent dat de klant om een UPDATE of een verandering in zijn huidige status verzoekt.

```
Container : SignIn.handleUserLoad(): Performing login: extn=2003 18:40:05:  
Container : [ClientServices] User: requestId='6e210ca9-5786-43bc-babf-64a397a6057f',  
</data>  
<event>PUT</event>  
<requestId>6e210ca9-5786-43bc-babf-64a397a6057f</requestId>  
<source>/finesse/api/User/2001</source>  
</Update>
```

De Fins REST API ontvangt dit verzoek van de cliënt. Vervolgens stuurt de API een **SetAgentStateReq** naar de CTI server.

```
%CCBU_http-8080-7-6-API_REQUEST: %[REQUEST_URL=User/2001][agent_id=2001]  
[request_identifier=6e210ca9-5786-43bc-babf-64a397a6057f][request_method=  
user.PUT][request_parameters= extension:2003 state:LOGIN]: Request from  
client to webservice api  
%CCBU_http-8080-7-6-REGISTER_API_STATS_OBJECT: %[resource_name=com.cisco.ccbu:  
category=WebAppStats,component0=User-[id]-PUT]: Registered new api stats object
```

for new request type.

```
%CCBU_http-8080-7-6-REQUEST_END: %[elapsed_time=8]: Request complete
%CCBU_pool-5-thread-4-6-MESSAGE_TO_CTI_SERVER: %[cti_message=Invoke id :20 ,
agentstate : 0, workmode : 0, reason code: -15532, forceflag :1, agentcapacity:
0, agenttext: 2003, agentid: 2001][cti_message_name=SetAgentStateReq]:
Message going to the backend cti server
```

De CTI Server ontvangt het verzoek.

```
Trace: AGENT_EVENT: ID=2001 Periph=5001 Ext=2003 Inst=2003 Sig=
Trace: SkgState=LOGIN SkgDuration=0 OverallState=NOT_READY OverallDuration=0
Reason=0
Trace: MRDID=1 NumTasks=0 MaxTaskLimit=1 AgtMode=1 AGTAvailabilityStatus=0
ICMAgtID=5001
Trace: SkTgtID=5001 SkGrpNo=0x0 SkGrpID=5006 NumLines=0 CurLine=0 ClientStatus=
0x0 Direction=0
```

Zodra de agent met een status van **NOT\_READY** is inlogd, verstuurt de CTI Server **AGENT\_STATE-EVENT** naar Finesse.

```
MsgType:AGENT_STATE_EVENT (MonitorID:0 PeripheralID:5001 SessionID:0x0
PeripheralType:EnterpriseAgent SkillGroupState:LOGIN StateDuration:0
SkillGroupNumber:85881 SkillGroupID:5000 SkillGroupPriority:0 AgentState:
NOT_READY EventReasonCode:0 MRDID:1 NumTasks:0 AgentMode:1 MaxTaskLimit:1
ICMAgentID:5001 AgentAvailabilityStatus:0 NumFltSkillGroups:0 Direction:0
ClientSignature:"AgentID:"2001" AgentExtension:"2003" AgentInstrument:"2003"
RemaskNumMasks:1 RemaskInstrument:"2003" RemaskExtension:"2003" RemaskCallId:
0xffffffff RemaskFunctionFlag:<0x38> <LogoutCodeReq,NotRdyCodeReq,WrapDataReq>
RemaskCallMask:<0x21000000> <MC,Emerg> RemaskAgentMask:<0x0a000000> <
Logout,Avail> )Trace: AGENT_EVENT: ID=2001 Periph=5001 Ext=2003 Inst=2003 Sig=
Trace: SkgState=LOGIN SkgDuration=0 OverallState=NOT_READY OverallDuration=0
Reason=0 Trace: MRDID=1 NumTasks=0 MaxTaskLimit=1 AgtMode=1
AGTAvailabilityStatus=0 ICMAgtID=5001
```

Dit is het webservicelogboek dat de gebeurtenis van de CTI Server heeft ontvangen. Onthoud dat je eerst het **RAW** bericht ziet van de CTI server en dan zie je het **gedecodeerde** bericht.

```
%CCBU_CTIMessageEventExecutor-0-6-DECODED_MESSAGE_FROM_CTI_SERVER: %[cti_message
=CTIAgentStateEvent [skillGroupState=0 (LOGIN), stateDuration=0, skillGroupNumber
=85881, skillGroupPriority=0, agentState=2 (NOT_READY), eventReasonCode=0,
numFltSkillGroups=0,CTIClientSignature=, agentID=2001, agentExtension=2003,
agentInstrument=2003, agentID_Long=null, duration=null, nextAgentState=null,
fltSkillGroupNumberList=[], fltSkill GroupIDList=[], fltSkillGroupPriorityList=[],
fltSkillGroupStateList=[]]CTIMessageBean [invokeID=null, msgID=30, timeTracker=
{"id":"AgentStateEvent", "CTI_MSG_RECEIVED":1366756808374,
"CTI_MSG_DISPATCH":1366756808375}, msgName=AgentStateEvent, deploymentType=CCE]]
[cti_response_time=1]: Decoded Message to Finesse from backend cti server
```

Nu Finesse de **AgentStateEvent** van de CTI server ontving, moet de gebeurtenis aan de Meldingsdienst worden **gepubliceerd** zodat de cliënt de UPDATE ontvangt. De enige manier waarop de beambte weet dat zijn of haar staat is veranderd is door deze **XMPP Event** te ontvangen. Finesse converteert de **AgentStateEvent** naar **XMPP** en stuurt de **XMPP** naar de Meldingsservice. Merk op dat de gebeurtenis een **PUT** is en dat de aanvraagID in de payload zit.

```
%CCBU_pool-5-thread-5-6-XMPP_PUBLISH_ASYNCHRONOUS: %[NodeId=/finesse/api/User/
2001] Payload=<Update><data><user><dialogs>/finesse/api/User/2001/Dialogs
</dialogs><extension>2003</extension><firstName>Mickey</firstName><lastName>
Mouse</lastName><loginId>2001</loginId><loginName>mmouse</loginName>
<reasonCodeId>-1</reasonCodeId><roles><role>Agent</role></roles><state>NOT_READY
</state><stateChangeTime>2013-04-23T22:40:08Z</stateChangeTime><teamId>5000
```

```
</teamId><teamName>Minnies_Team</teamName><uri>/finesse/api/User/2001
</uri></user></data><event>PUT</event><requestId>6e210ca9-5786-43bc-babf-
64a397a6057f </requestId><source>/finesse/api/User/2001</source></Update>]:
Publishing XMPP Message Asynchronously
```

Hier ontvangt de Aanmeldingsdienst de UPDATE. Hoewel het bericht zegt dat **pakketjes niet naar JID zijn verzonden**, wordt het bericht dat een gebeurtenis is gepubliceerd naar de gebruiker verstuurd.

```
RoutingTableImpl: Failed to route packet to JID: 2001@uccefinesse138.vmload.cvp/
User packet: <message from="pubsub.uccefinesse138.vmload.cvp" to=
"2001@uccefinesse138.vmload.cvp/ User" id="/finesse/api/User/
2001__2001@uccefinesse138.vmload.cvp__VI1B2"><event xmlns=
"http://jabber.org/protocol/pubsub#event"><items node="/finesse/api/User/2001">
<item id="1su0Keff8M2irdS"><notification xmlns="http://jabber.org/protocol/pubsub">
&lt;Update&gt;
```

Hier volgt de essentie van de boodschap:

```
&lt;data&gt;
&lt;user&gt;
&lt;dialogs&gt;/finesse/api/User/2001/Dialogs&lt;/dialogs&gt;
&lt;extension&gt;2003&lt;/extension&gt;
&lt;firstName&gt;Mickey&lt;/firstName&gt;
&lt;lastName&gt;Mouse&lt;/lastName&gt;
&lt;loginId&gt;2001&lt;/loginId&gt;
&lt;loginName&gt;mmouse&lt;/loginName&gt;
&lt;reasonCodeId&gt;-1&lt;/reasonCodeId&gt;
&lt;roles&gt;
&lt;role&gt;Agent&lt;/role&gt;
&lt;/roles&gt;
&lt;state&gt;NOT_READY&lt;/state&gt;
&lt;stateChangeTime&gt;2013-04-23T22:40:08Z&lt;/stateChangeTime&gt;
&lt;teamId&gt;5000&lt;/teamId&gt;
&lt;teamName&gt;Minnies_Team&lt;/teamName&gt;
&lt;uri&gt;/finesse/api/User/2001&lt;/uri&gt;
&lt;/user&gt;
&lt;/data&gt;
&lt;event&gt;PUT&lt;/event&gt;
&lt;requestId&gt;6e210ca9-5786-43bc-babf-64a397a6057f</requestId&gt;
&lt;source&gt;/finesse/api/User/2001&lt;/source&gt;
&lt;/Update&gt;</notification></item></items></event></message>
```

Zoals eerder, wordt het XMPP bericht ontvangen door de cliënt en geleverd aan de Gadget van de cliënt. Merk op dat de klant de gebeurtenis met de oorspronkelijke aanvraagID in het bericht ontvangt.

```
Returned with status=202, content=''18:40:05: Container : [ClientServices]
MasterPublisher._eventHandler() - Received event on node '/finesse/api/User/
2001': <Update>
<data>
<user>
<dialogs>/finesse/api/User/2001/Dialogs</dialogs>
<extension>2003</extension>
<firstName>Mickey</firstName>
<lastName>Mouse</lastName>
<loginId>2001</loginId>
<loginName>mmouse</loginName>
<reasonCodeId>-1</reasonCodeId>
<roles>
<role>Agent</role>
```



```
</roles>
<state>NOT_READY</state>
<stateChangeTime>2013-04-23T22:40:08Z</stateChangeTime>
<teamId>5000</teamId>
<teamName>Minnies_Team</teamName>
<uri>/finesse/api/User/2001</uri>
</user>
</data>
<event>PUT</event>
<requestId>6e210ca9-5786-43bc-babf-64a397a6057f</requestId>
<source>/finesse/api/User/2001</source>
</Update>
```

De client is inlogd.

```
Container : SignIn._triggerLoggedIn(): Successfully logged in!18:40:05
```

## Uitlogcodes, reden, telefoon, boek

Nu moet de klant agent-specifieke gegevens, zoals Logout codes, Moedercodes en Telefoonboek terugkrijgen. Hier is het verzoek om die informatie aan de cliënt.

```
Container : SignIn._triggerLoggedIn(): Successfully logged in!18:40:05:
Container : [ClientServices] Dialogs: requestId='undefined', Making REST
request: method=GET, url='/finesse/api/User/2001/Dialogs?nocache=
1366756805355?
```

```
18:40:05: Container : [ClientServices] User: requestId='undefined',
Making REST request: method=GET, url='/finesse/api/User/2001/ReasonCodes?
category=LOGOUT&nocache=1366756805356'18:40:05: Container : [ClientServices]
User: requestId='undefined', POST_DATA=''18:40:05: Container : _displayUserData
(): User's current state is: NOT_READY
```

```
'18:40:05: Container : [ClientServices] User: requestId='undefined', Making REST
request: method=GET, url='/finesse/api/User/2001/ReasonCodes?category=NOT_READY&
nocache=1366756805358
```

```
18:40:05: Container : [ClientServices] User: requestId='undefined', POST_DATA=
''18:40:05: Header : The client logger has been initialize for the header
18:40:05: Header : _displayUserData(): User's current state is: NOT_READY
```

```
18:40:05: Header : Container._initGadgetContainer(): Initializing gadget
container.
```

```
18:40:05: Header : FailoverMonitor.startListening(): Listening for triggers
```

```
18:40:05: Header : PageServices.stopTimeoutPoller(): Cancelling connection
timeout and poller...
```

```
18:40:05: Header : [ClientServices] id=2001: TypeError: 'this._listenerCallback
[...].callback' is null or not an object
```

Dezelfde logica is van toepassing op deze verzoeken. Houd in gedachten dat de codes Finesse Reason en PhoneBook in de Finse database zijn opgeslagen, niet in UCCE.