

# A seleção de gateway ótima de AnyConnect pesquisa defeitos o guia

## Índice

[Introdução](#)

[Como OGS trabalha?](#)

[Esconderijo OGS](#)

[Determinação do lugar](#)

[Cenários de falha](#)

[Quando a Conectividade ao gateway for perdida](#)

[Resumo após uma suspensão](#)

[O tamanho de janela TCP Atrasar-ACK seleciona o gateway incorreto](#)

[Exemplo típico do usuário](#)

[Pesquise defeitos OGS](#)

[Etapa 1. Cancele o escondimento OGS a fim forçar uma reavaliação](#)

[Etapa 2. Capture as pontas de prova do server durante a tentativa de conexão](#)

[Etapa 3. Verifique o gateway selecionado por OGS](#)

[Etapa 4. Valide os cálculos OGS executados por AnyConnect](#)

[Análise](#)

[Q&A](#)

## Introdução

Este documento descreve como pesquisar defeitos edições com seleção de gateway ótima (OGS). OGS é uma característica que possa ser usada a fim determinar que gateway tem o mais baixo Round Trip Time (RTT) e o conectar a esse gateway. Um pode usar a característica OGS a fim minimizar a latência para o tráfego do Internet sem intervenção de usuário. Com OGS, o Cliente de mobilidade Cisco AnyConnect Secure (AnyConnect) identifica e seleciona que fixam o gateway são os melhores para a conexão ou a reconexão. OGS começa em cima da primeira conexão ou em cima de uma reconexão pelo menos quatro horas após a desconexão precedente. Mais informação pode ser encontrada no [guia de administrador](#).

Dica: OGS trabalha melhor com o cliente o mais atrasado de AnyConnect e versão de software ASA 9.1(3) \* ou mais tarde.

## Como OGS trabalha?

Uma solicitação de ping simples do Internet Control Message Protocol (ICMP) não trabalha porque muitos Firewall adaptáveis da ferramenta de segurança de Cisco (ASA) são configurados para obstruir pacotes ICMP a fim impedir a descoberta. Em lugar de, o cliente envia três pedidos HTTP/443 a cada final do cabeçalho que aparece em uma  **fusão de**  todos os perfis. Estas provas HTTP estão referidas enquanto OGS sibila nos logs, mas, como explicado mais cedo, não é ping ICMP. A fim assegurar-se de que a conexão a (com referência a) não tome demasiado por muito

tempo, OGS seleciona o gateway precedente à revelia se não recebe nenhuns resultados do sibilo OGS dentro de sete segundos. (Procure **resultados do sibilo OGS no log.**)

Nota: AnyConnect deve enviar um pedido do HTTP a 443, porque a resposta própria é importante, não uma resposta bem sucedida. Infelizmente, o reparo para a manipulação do proxy envia todos os pedidos como o HTTPS. Veja a identificação de bug Cisco [CSCtg38672](#) - OGS deve sibilar com pedidos do HTTP.

Nota: Se não há nenhum final do cabeçalho no esconderijo, AnyConnect envia primeiramente um pedido do HTTP a fim determinar se há um Proxy de autenticação, e se pode segurar o pedido. É somente depois esta solicitação inicial que começa os sibilos OGS a fim sondar o server.

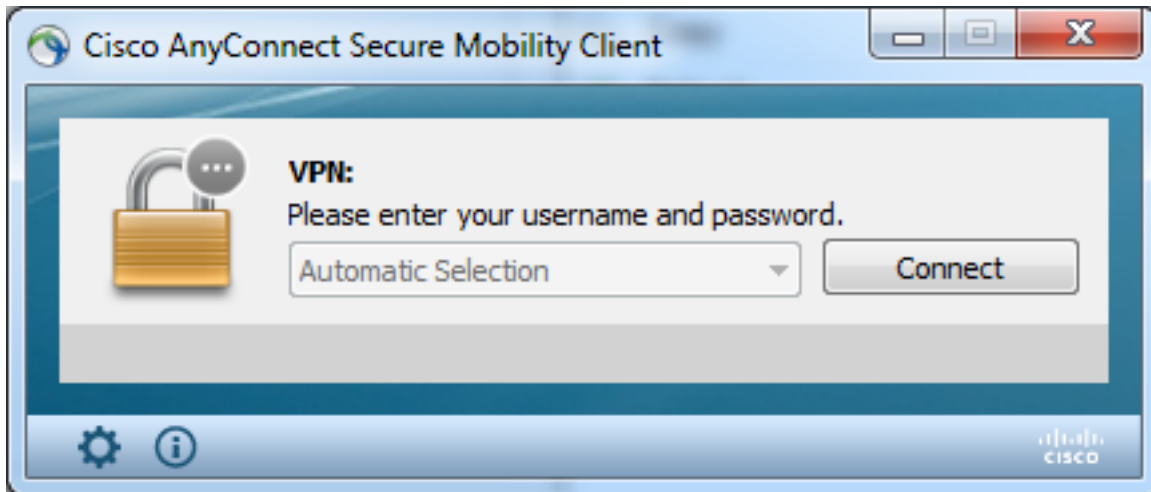
- OGS determina o lugar do usuário baseado na informação de rede, tal como o sufixo do Domain Name System (DNS) e o endereço IP de Um ou Mais Servidores Cisco ICM NT do servidor DNS. Os resultados RTT, junto com este lugar, são armazenados no esconderijo OGS.
- As entradas do lugar OGS são postas em esconderijo por 14 dias. A identificação de bug Cisco [CSCtk66531](#) foi arquivada para fazer a estes configuráveis pelo usuário dos ajustes.
- OGS não está executado outra vez deste lugar até 14 dias depois que a entrada do lugar é posta em esconderijo primeiramente. Durante este tempo, usa a entrada oculta e os RTT determinado para esse lugar. Isto significa que quando AnyConnect começa outra vez, não executa OGS outra vez; em lugar de, usa a ordem ótima do gateway no esconderijo para esse lugar. Nos logs diagnósticos da ferramenta de relatório de AnyConnect (DARDO), esta mensagem é considerada:

```
*****  
Date : 10/04/2013  
Time : 14:00:44  
Type : Information  
Source : acvpnui  
  
Description : Function: ClientIfcBase::startAHS  
File: .\ClientIfcBase.cpp  
Line: 2785  
OGS was already performed, previous selection will be used.
```

- O RTT é determinado com uma troca TCP à porta do secure sockets layer (SSL) do gateway a que o usuário tentará conectar como especificado pela entrada de host no perfil de AnyConnect.

Nota: Ao contrário do HTTP-sibilo, que faz um cargo simples HTTP e indica então o RTT e o resultado, as computações OGS são levemente mais complicadas. AnyConnect envia três pontas de prova para cada server, e calcula o atraso entre o HTTP SYN que manda e o FIN/ACK para cada um destes sonda. Usa então o mais baixo dos deltas a fim comparar os server e fazer sua seleção. Assim, mesmo que os HTTP-sibilos fossem uma indicação razoavelmente boa de que o server o AnyConnect escolherá, não puderam necessariamente registrar. Há mais informação sobre esta no resto do documento.

- Atualmente, OGS executa somente as verificações se o usuário sai de uma suspensão, e o ponto inicial esteve excedido. OGS não conecta a um ASA diferente se o ASA o usuário é conectado aos impactos ou se torna não disponível. OGS contacta somente os servidores primários no perfil a fim determinar ótimo.
- Uma vez que o perfil do cliente OGS está transferido, quando o usuário reinicia o cliente de AnyConnect, a opção para selecionar outros perfis será esmaecida para fora como mostrado aqui:



Mesmo se a máquina do usuário tem o múltiplo outros perfis não poderão selecionar alguns deles até que OGS estado disbaled.

## Esconderijo OGS

Uma vez que o cálculo é, os resultados armazenados no arquivo **preferences\_global**. Houve umas edições com estes dados que não estão sendo armazenados no arquivo antes.

Refira a identificação de bug Cisco [CSCtj84626](https://cisco.com/cisco/webbugtools/bugdetails/CSCtj84626) para mais detalhes.

## Determinação do lugar

OGS que põe em escondერიjo trabalhos em uma combinação do domínio de DNS e dos endereços IP do servidor dos DN individuais. Trabalha como segue:

- O lugar A tem um domínio de DNS de **locationa.com**, e dois endereços IP de Um ou Mais Servidores Cisco ICM NT do servidor DNS - **ip1** e **ip2**. Cada combinação domain/IP cria uma chave do escondერიjo esses pontos a uma entrada de cache OGS. Por exemplo: **locationa.com|ip1 -> ogscache1locationa.com|ip2 -> ogscache1**
- Se AnyConnect conecta então a uma rede físico-diferente, o mesmo acúmulo de combinações domain/IP está criado e verificado contra a lista posta em escondერიjo. Se há algum fósforo de todo, esse valor do escondერიjo OGS está usado, e o cliente é considerado ainda estar no **lugar A**.

## Cenários de falha

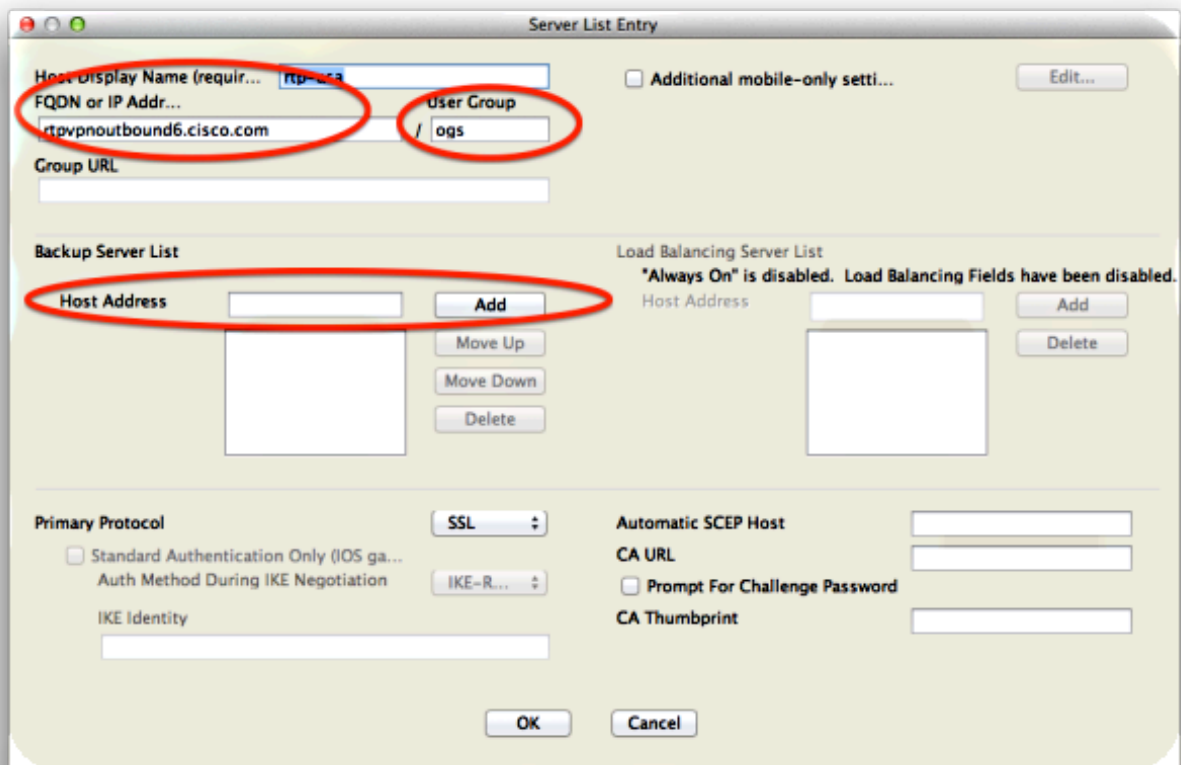
Estão aqui alguns cenários de falha que os usuários puderam encontrar:

## Quando a Conectividade ao gateway for perdida

Quando OGS está usado, se a Conectividade ao gateway a que os usuários estão conectados está perdida, a seguir AnyConnect conecta aos server no listandnot do servidor de backup ao host seguinte OGS. O ordem de operação é como segue:

1. OGS contacta somente os servidores primários a fim determinar ótimo.
2. Uma vez que determinado, o algoritmo da conexão é:  
Tentativa de conectar ao server ótimo. Se isso falha, tente a lista do servidor de backup do server ótimo. Se isso falha, tente cada server que permanece na lista da seleção OGS, pedido por sua seleção resulta.

Nota: Quando o administrador configura a lista do servidor de backup, o editor atual do perfil permite somente que o administrador incorpore o nome de domínio totalmente qualificado (FQDN) para o servidor de backup, mas não o grupo de utilizadores como é possível para o servidor primário:



A identificação de bug Cisco [CSCud84778](#) foi arquivada a fim corrigir esta, mas a URL completa deve ser incorporada ao campo do endereço de host para o servidor de backup, e deve trabalhar: `https://<ip-address>/usergroup`.

## Resumo após uma suspensão

Para que OGS seja executado depois que um resumo, AnyConnect deve ter tido uma conexão estabelecida quando a máquina esteve posta para dormir. OGS depois que um resumo é executado somente depois que o teste do ambiente de rede ocorre, que está significado confirmar que a conectividade de rede está disponível. Este teste inclui uma Conectividade DNS a mais

subtest.

Contudo, se o servidor DNS deixa cair o tipo pedidos A com um endereço IP de Um ou Mais Servidores Cisco ICM NT no campo da pergunta, ao contrário da resposta com o “nome não encontrado” (o caso mais comum, encontrado sempre durante testes), a seguir da identificação de bug Cisco [CSCti20768](#) “pergunta DNS do tipo A para o endereço IP de Um ou Mais Servidores Cisco ICM NT, deve ser o PTR para evitar o intervalo” aplica-se.

## O tamanho de janela TCP Atrasar-ACK seleciona o gateway incorreto

Quando as versões ASA do que a versão 9.1(3) são usadas mais cedo, as captações no cliente mostram um atraso persistente na saudação de SSL. O que é observado é que o cliente envia seu ClientHello, a seguir o ASA envia seu ServerHello. Isto é seguido normalmente por uma mensagem do certificado (pedido do certificado opcional) e pela mensagem de ServerHelloDone. A anomalia é dupla:

1. O ASA não envia imediatamente a mensagem do certificado após o ServerHello. O tamanho da janela de cliente é 64,860 bytes, que é mais do que bastante para guardar a resposta inteira do ASA.
2. O cliente não faz ACK o ServerHello imediatamente, assim que o ASA retransmite o ServerHello após ~120ms, que no ponto o cliente ACK os dados. A mensagem do certificado é enviada então. É quase como se o cliente espera mais dados.

Isto acontece devido à interação entre o lento-[início TCP](#) e o [TCP ATRASAR-ACK](#). Antes da versão ASA 9.1(3), o ASA usa um tamanho de janela do lento-início de 1, visto que o cliente do Windows usa um valor atrasar-ACK de 2. Isto significa que o ASA envia somente um pacote de dados até que obtenha um ACK, mas igualmente significa que o cliente não envia um ACK até que receba dois pacotes de dados. Os tempos ASA para fora depois que 120ms e retransmite o ServerHello, depois do qual o cliente ACK os dados e a conexão continua. Este comportamento foi mudado pela identificação de bug Cisco [CSCug98113](#) de modo que o ASA usasse um tamanho de janela lento do começo de 2 à revelia em vez de 1.

Isto pode impactar o cálculo OGS quando:

- Os gateways diferentes executam versões ASA diferentes.
- Os clientes têm os tamanhos de janela atrasar-ACK diferentes.

Em tais situações, o atraso introduzido pelo atrasar-ACK podia ser suficiente para fazer com que o cliente selecione o ASA errado. Se este valor difere entre o cliente e o ASA, poderia ainda haver uns problemas. Em tais situações, a ação alternativa é ajustar o tamanho de janela atrasado dos reconhecimentos.

## Windows

1. Comece o **editor de registro**.
2. Identifique o GUID da relação em que você quer desabilitar o atrasar-ACK. A fim fazer isto, navegue a:  
**HKEY\_LOCAL\_MACHINE > SOFTWARE > Microsoft > Windows NT > Versão atual > NetworkCards > (número)**.  
Olhe cada número alistado sob NetworkCards. No lado direito, a descrição deve alistar a relação (por exemplo, Intel (R) link wireless 5100AGN de WiFi) e o Nome de serviço devem alistar o GUID correspondente.
3. Encontre e clique então esta subchave do registro:

HKEY\_LOCAL\_MACHINE \ SISTEMA \ CurrentControlSet \ serviços \ Tcpip \ parâmetros \ relações \ <Interface GUID>

4. No menu da edição, o ponto a novo, e clica então o **valor DWORD**.
5. Nomeie o valor novo **TcpAckFrequency**, e atribua-lhe um valor de **1**.
6. Pare o editor de registro.
7. Reinicie Windows para que esta mudança tome o efeito.

Nota: A identificação de bug Cisco [CSCum19065](#) foi arquivada para fazer parâmetros de ajuste TCP configuráveis no ASA.

## Exemplo típico do usuário

O caso o mais de utilização comum é quando um usuário executa em casa OGS a primeira vez, ele grava os ajustes DNS e os resultados do sibilho OGS no esconderijo (padrões a um intervalo de 14-dia). Quando o usuário retorna em casa a próxima noite, OGS detecta os mesmos ajustes DNS, encontra-os no esconderijo, e salta-o o teste de ping OGS. Mais tarde, quando o usuário vai a um hotel ou a um restaurante que ofereça o serviço de Internet, OGS detecta ajustes diferentes DNS, executa os testes de ping OGS, seleciona o melhor gateway, e grava os resultados no esconderijo.

O processamento é idêntico quando recomeça de um estado suspenso ou hibernado, se os ajustes do resumo OGS e de AnyConnect permitem ele.

## Pesquisa defeitos OGS

### Etapa 1. Cancele o esconderijo OGS a fim forçar uma reavaliação

A fim cancelar os OGS põem em esconderijo e reavaliam o RTT para gateways disponíveis, suprimem simplesmente das preferências globais de AnyConnect arquivam do PC. O lugar do arquivo varia baseado no operating system (OS):

- Windows Vista e Windows 7

C:\ProgramData\Cisco\Cisco AnyConnect Secure Mobility Client\preferences\_global.xml  
Note: in older client versions it used to be stored in C:\ProgramData\Cisco\Cisco AnyConnect VPN Client

- Windows XP

C:\Documents and Settings\AllUsers\Application Data\Cisco\Cisco AnyConnect VPN Client\preferences\_global.xml

- Mac OS X

/opt/cisco/anyconnect/.anyconnect\_global  
Note: with older versions of the client it used to be /opt/cisco/vpn..

- Linux

/opt/cisco/anyconnect/.anyconnect\_global  
Note: with older versions of the client it used to be /opt/cisco/vpn..

### Etapa 2. Capture as pontas de prova do server durante a tentativa de conexão

1. Comece Wireshark na máquina do teste.
2. Comece uma tentativa de conexão em AnyConnect.
3. Pare a captação de Wireshark uma vez que a conexão está completa. Dica: Desde que a captação é usada somente a fim de testar OGS, é o melhor parar a captação assim que AnyConnect selecionar um gateway. É o melhor não atravessar uma tentativa de conexão completa, porque aquele pode se nublar a captura de pacote de informação.

### Etapa 3. Verifique o gateway selecionado por OGS

A fim de verificar porque OGS selecionou um gateway particular, termine estas etapas:

1. Inicie uma nova conexão.
2. Execute o DARDO de AnyConnect:  
Lance **AnyConnect**, e clique **avançado**. Clique **diagnósticos**. Clique em **Next**. Clique em **Next**.
3. Examine os resultados do DARDO encontrados no **arquivo** recém-criado no desktop.  
Navegue ao **Cliente de mobilidade Cisco AnyConnect Secure > ao AnyConnect.txt**.

Note o tempo onde as pontas de prova OGS começaram para um servidor particular deste DARDO registram:

\*\*\*\*\*

```
Date : 10/04/2013
Time : 14:21:27
Type : Information
Source : acvpnu
```

```
Description : Function: CHeadendSelection::CSelectionThread::Run
File: .\AHS\HeadendSelection.cpp
Line: 928
OGS starting thread named gw2.cisco.com
```

\*\*\*\*\*

Geralmente devem ser em torno do mesmo tempo, mas caso que as captações são grandes, o selo de tempo ajuda a reduzir para baixo que os pacotes são as provas HTTP e qual são as tentativas de conexão reais.

Uma vez que AnyConnect envia três pontas de prova ao server, esta mensagem está gerada com os resultados para cada um das pontas de prova:

\*\*\*\*\*

```
Date : 10/04/2013
Time : 14:31:37
Type : Information
Source : acvpnu
```

```
Description : Function: CHeadendSelection::CSelectionThread::logThreadPingResults
File: .\AHS\HeadendSelection.cpp
Line: 1137
OGS ping results for gw2.cisco.com: (219 218 132 )
```

\*\*\*\*\* É importante pagar a atenção a estes três

valores, porque devem combinar os resultados da captação.

Procure a mensagem que contém do “o \*\*\* dos resultados da seleção \*\*\* OGS” a fim considerar o RTT avaliado, e se a tentativa de conexão a mais recente era o resultado de um RTT posto em esconderijo ou de um cálculo novo.

Aqui está um exemplo: \*\*\*\*\*

```
Date       : 10/04/2013
Time       : 12:29:38
Type       : Information
Source     : vpnui

Description : Function: CHeadendSelection::logPingResults
File:      .\AHS\HeadendSelection.cpp
Line:      589
*** OGS Selection Results ***
OGS performed for connection attempt. Last server: 'gw2.cisco.com'

Results obtained from OGS cache. No ping tests were performed.

Server Address      RTT (ms)
gw1.cisco.com       302
gw2.cisco.com       132 <===== As seen, 132 was the lowest delay
of the three probes from the previous DART log
gw3.cisco.com       506
gw4.cisco.com       877

Selected 'gw2.cisco.com' as the optimal server.

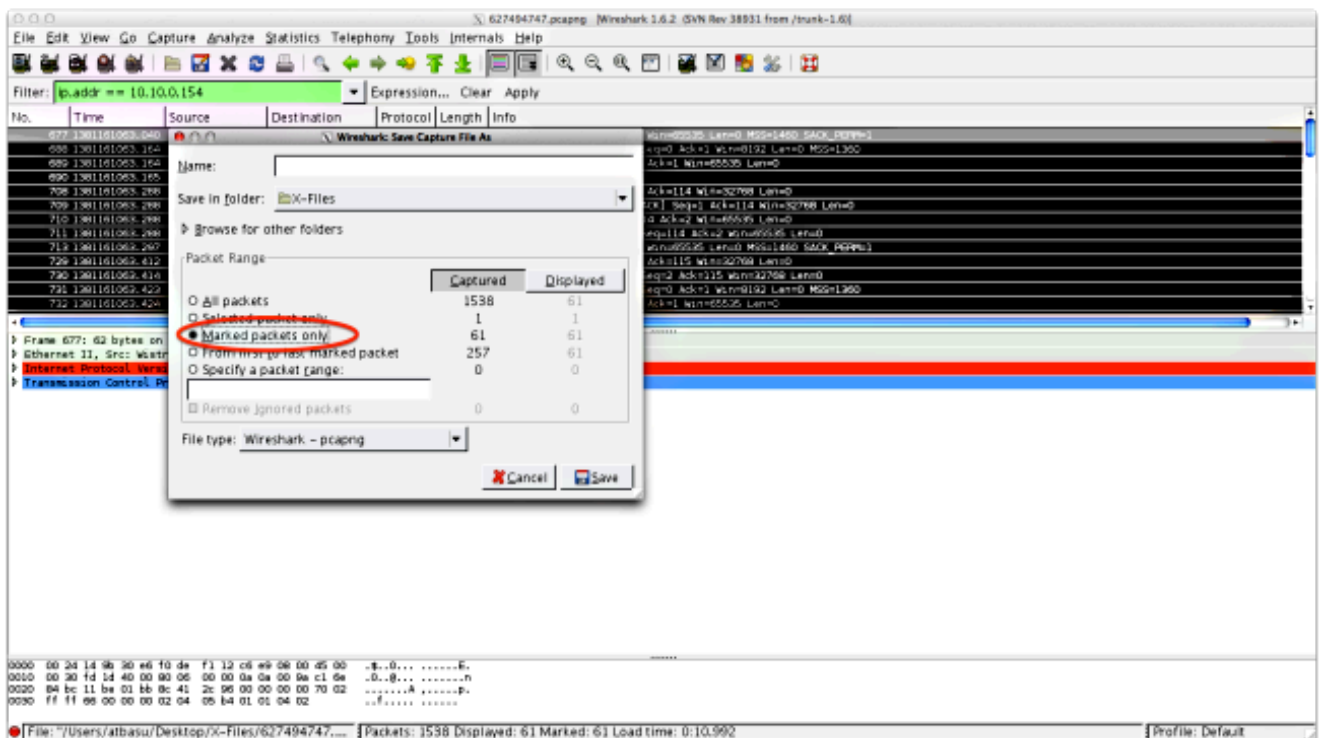
*****
```

## Etapa 4. Valide os cálculos OGS executados por AnyConnect

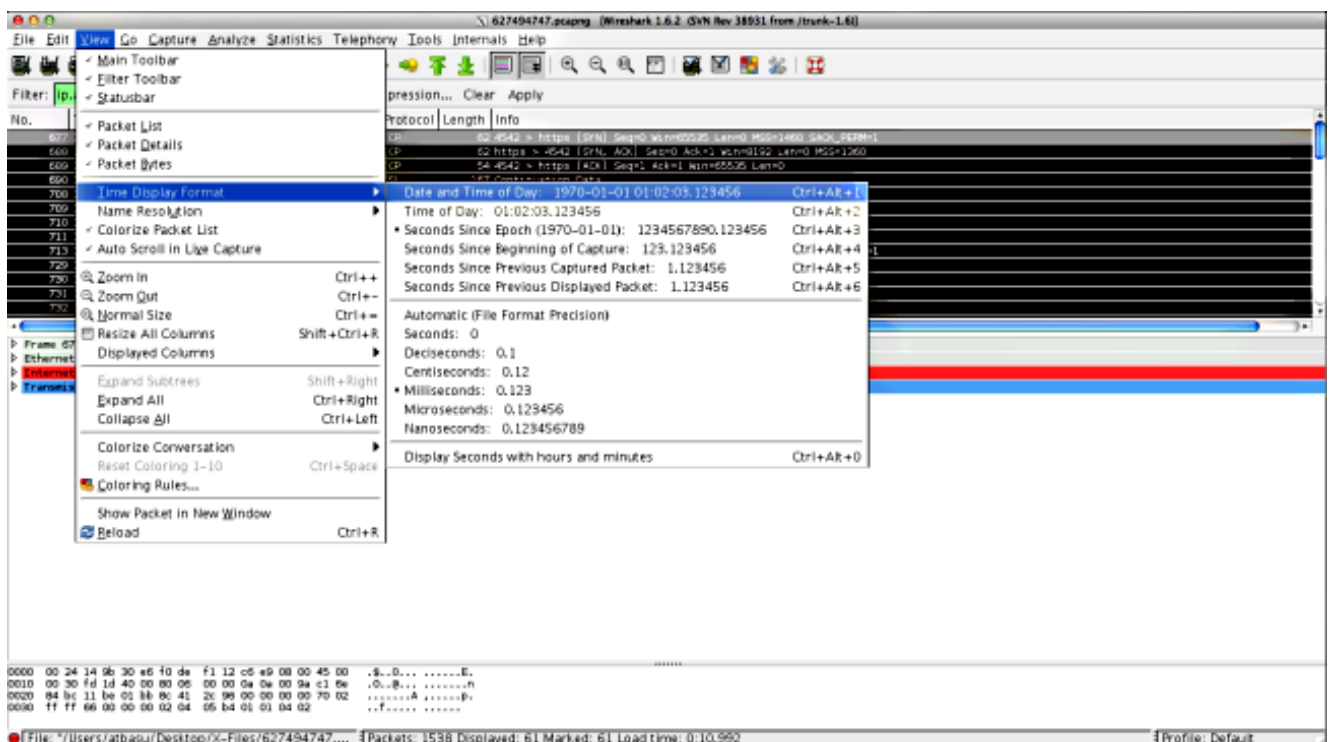
Inspecione a captação para o TCP/SSL sonda usado a fim calcular o RTT. Veja quanto tempo o pedido HTTPS toma sobre uma única conexão de TCP. Cada pedido da ponta de prova deve usar uma conexão de TCP diferente. A fim fazer isto, abra a captação em Wireshark, e repita estas etapas para cada um dos server:

1. Use o **filtro ip.addr** a fim isolar os pacotes enviados a cada um dos server em sua própria captação. A fim fazer isto, para navegar para editar, e MarkAll seletor **indicou pacotes**. Então navegue ao arquivo > salvar como, **selecione a opção de Markedpackets somente**, e clique a salvaguarda:





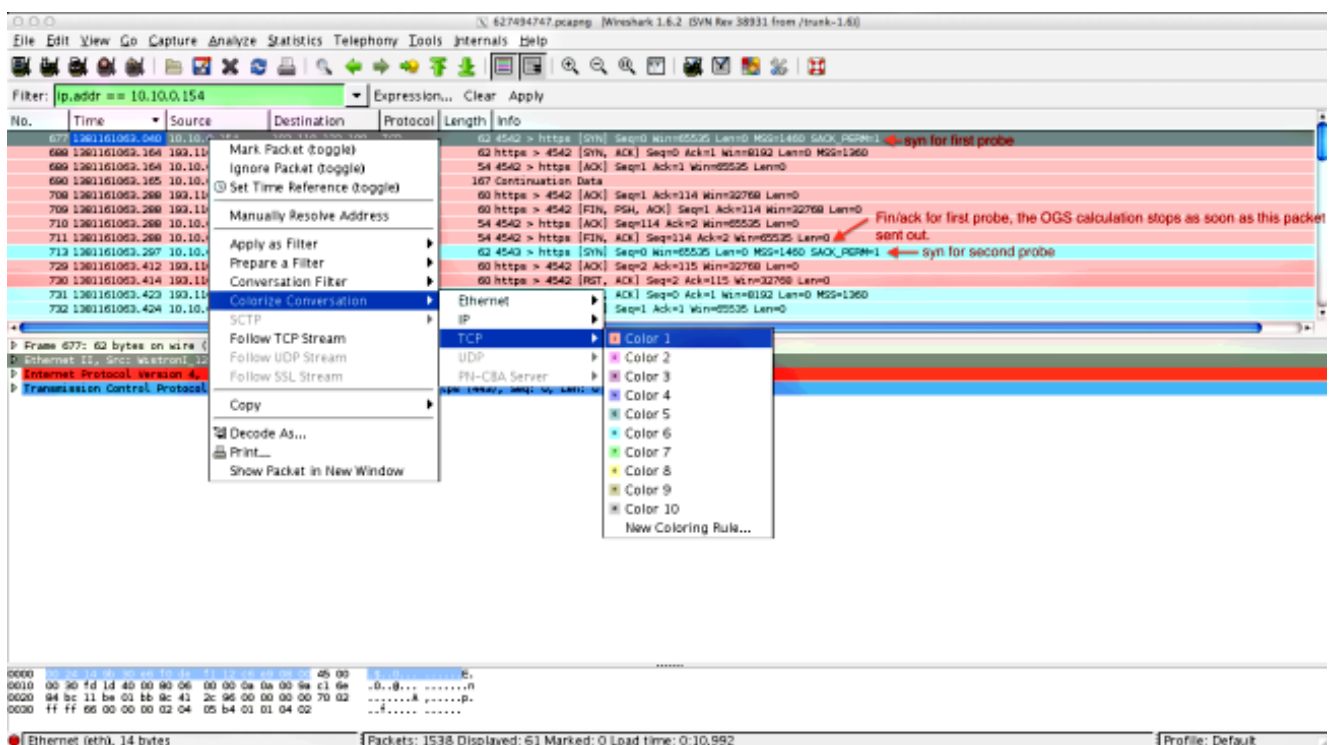
2. Nesta captação nova, navegue para ver > formato de exibição > data e Time Of Day do tempo:



3. Identifique o primeiro pacote SYN HTTP nesta captação que foi enviada quando a ponta de prova OGS foi enviada baseada nos logs do DARDO como identificada em etapa 3.3.2. É importante recordar que, para o primeiro server, o primeiro pedido do HTTP não é uma ponta de prova do server. É fácil confundir o primeiro pedido por uma ponta de prova do server, e chega assim nos valores completamente diferentes de que OGS relata. Este problema é destacado aqui:

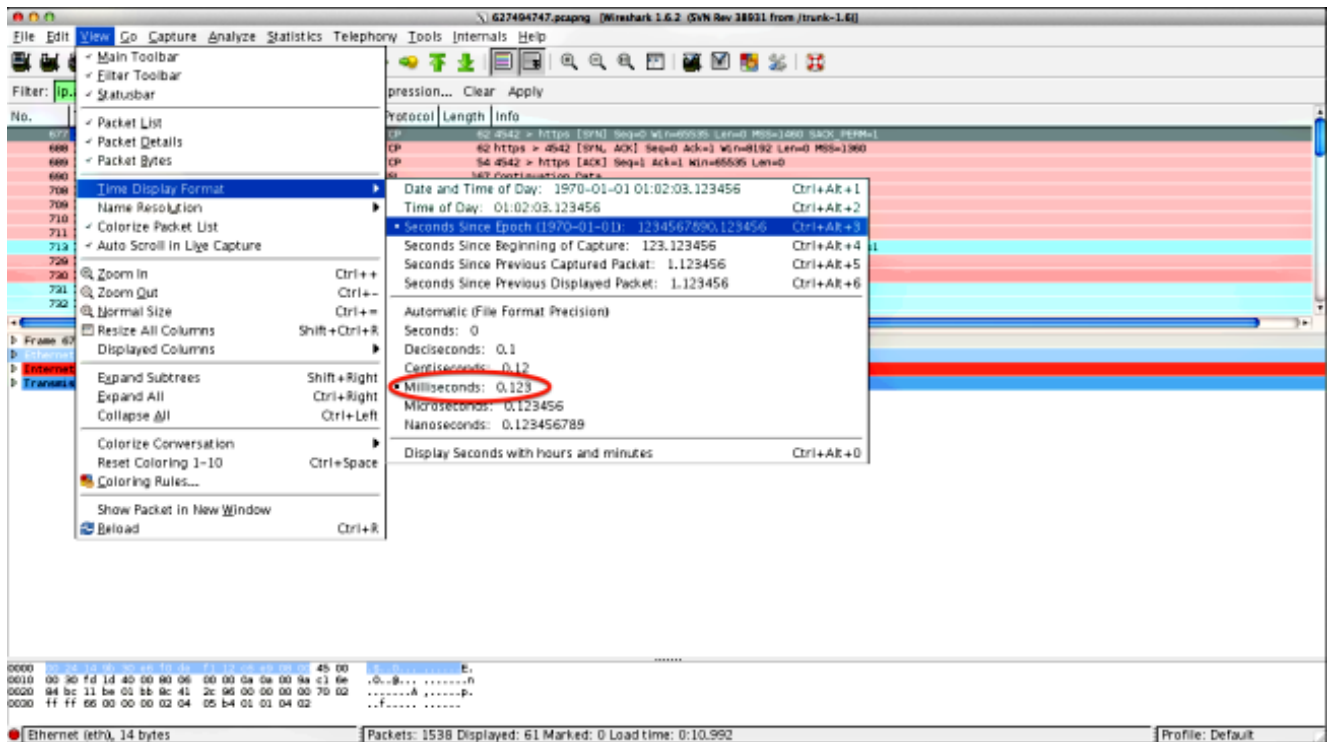
677	2013-10-07	11:51:03.040834	10.10.0.154		10.10.0.154	TCP	62	4542 > https [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1
689	2013-10-07	11:51:03.164883	10.10.0.154	Test HTTP Connection	10.10.0.154	TCP	54	4542 > https [ACK] Seq=1 Ack=1 Win=65535 Len=0
690	2013-10-07	11:51:03.165061	10.10.0.154		10.10.0.154	SSL	167	Continuation Data
710	2013-10-07	11:51:03.288837	10.10.0.154		10.10.0.154	TCP	54	4542 > https [ACK] Seq=114 Ack=2 Win=65535 Len=0
711	2013-10-07	11:51:03.288937	10.10.0.154		10.10.0.154	TCP	54	4542 > https [FIN, ACK] Seq=114 Ack=2 Win=65535 Len=0
713	2013-10-07	11:51:03.297522	10.10.0.154		10.10.0.154	TCP	62	4543 > https [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1
732	2013-10-07	11:51:03.424015	10.10.0.154		10.10.0.154	TCP	54	4543 > https [ACK] Seq=1 Ack=1 Win=65535 Len=0
734	2013-10-07	11:51:03.424384	10.10.0.154		10.10.0.154	TLSv1	131	Client Hello
762	2013-10-07	11:51:03.552735	10.10.0.154	OQS Test 1	10.10.0.154	TCP	54	4543 > https [ACK] Seq=78 Ack=1486 Win=65535 Len=0
763	2013-10-07	11:51:03.553816	10.10.0.154		10.10.0.154	TLSv1	368	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Mess
779	2013-10-07	11:51:03.747197	10.10.0.154		10.10.0.154	TLSv1	192	Application Data
792	2013-10-07	11:51:03.874861	10.10.0.154		10.10.0.154	TCP	54	4543 > https [ACK] Seq=530 Ack=1850 Win=65172 Len=0
793	2013-10-07	11:51:03.876186	10.10.0.154		10.10.0.154	TCP	54	4543 > https [FIN, ACK] Seq=530 Ack=1850 Win=65172 Len=0
794	2013-10-07	11:51:03.877037	10.10.0.154		10.10.0.154	TCP	62	lamer-1e > https [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1
809	2013-10-07	11:51:04.001156	10.10.0.154		10.10.0.154	TCP	54	lamer-1e > https [ACK] Seq=1 Ack=1 Win=65535 Len=0
810	2013-10-07	11:51:04.003693	10.10.0.154		10.10.0.154	TLSv1	163	Client Hello
827	2013-10-07	11:51:04.127077	10.10.0.154	OQS Test 2	10.10.0.154	TLSv1	101	Change Cipher Spec, Encrypted Handshake Message
828	2013-10-07	11:51:04.129515	10.10.0.154		10.10.0.154	TLSv1	192	Application Data
844	2013-10-07	11:51:04.254841	10.10.0.154		10.10.0.154	TCP	54	lamer-1e > https [ACK] Seq=295 Ack=444 Win=65093 Len=0
845	2013-10-07	11:51:04.254869	10.10.0.154		10.10.0.154	TCP	54	lamer-1e > https [FIN, ACK] Seq=295 Ack=444 Win=65093 Len=0
846	2013-10-07	11:51:04.255775	10.10.0.154		10.10.0.154	TCP	62	gds-adpflw-db > https [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_P
856	2013-10-07	11:51:04.382426	10.10.0.154		10.10.0.154	TCP	54	gds-adpflw-db > https [ACK] Seq=1 Ack=1 Win=65535 Len=0
857	2013-10-07	11:51:04.382941	10.10.0.154		10.10.0.154	TLSv1	163	Client Hello
866	2013-10-07	11:51:04.510362	10.10.0.154	OQS Test 3	10.10.0.154	TLSv1	101	Change Cipher Spec, Encrypted Handshake Message
867	2013-10-07	11:51:04.512581	10.10.0.154		10.10.0.154	TLSv1	192	Application Data
895	2013-10-07	11:51:04.639659	10.10.0.154		10.10.0.154	TCP	54	gds-adpflw-db > https [ACK] Seq=295 Ack=444 Win=65093 Len=0
896	2013-10-07	11:51:04.640162	10.10.0.154		10.10.0.154	TCP	54	gds-adpflw-db > https [FIN, ACK] Seq=295 Ack=444 Win=65093 Len=0

4. A fim identificar mais facilmente cada um das pontas de prova, clicar com o botão direito no **HTTP SYN** para a primeira ponta de prova, e seleccione então a conversação de Colorize como mostrado aqui:



Repita este processo para os SYN em todas as pontas de prova. Segundo as indicações da imagem anterior, as primeiras duas pontas de prova são descritas em cores diferentes. A vantagem de colorizing as conversações TCP é manchar facilmente retransmissões ou outras tais estranhezas pela ponta de prova.

5. A fim mudar o indicador do tempo, navegue para **ver > formato de exibição > segundos do tempo desde a época**:



Selecione **milissegundos**, porque aquele é o nível da precisão que OGS usa.

- Calcule a diferença de horário entre o HTTP SYN e o FIN/ACK, segundo as indicações do diagrama da repetição de etapa 4. este processo para cada um das três pontas de prova, e compare os valores àqueles mostrados no DARDO entra etapa 3.3.3.

## Análise

Se depois que a análise das captações os valores determinados RTT está calculada e comparada aos valores considerados nos logs do DARDO e tudo está encontrado para combinar acima, mas ainda parece como o gateway errado está sendo selecionado, a seguir é devido a um de dois problemas:

- Há uma edição no final do cabeçalho. Se este é o caso, pôde haver retransmissões demais de um final do cabeçalho particular, ou todas as outras tais estranhezas vistas nas pontas de prova. Uma análise mais próxima da troca é exigida.
- Há um problema com o provedor de serviço do Internet (ISP). Se este é o caso, pôde haver uma fragmentação ou uns grandes atrasos vista para um final do cabeçalho particular.

## Q&A

**P:** OGS trabalha com função de balanceamento de carga?

**R:** Sim. OGS está somente ciente do nome do mestre do conjunto, e dos usos que a fim julgar o final do cabeçalho o mais próximo.

**P:** OGS trabalha com os ajustes do proxy definidos no navegador?

**R:** OGS não apoia auto auto (PAC) arquivos do proxy ou da configuração do proxy, mas apoia um

servidor proxy duro-codificado. Como tal, a operação OGS não ocorre. O mensagem de registro relevante é: **“OGS não será executado porque a detecção automática do proxy é configurada.”**