トラブルシューティングおよびベスト プラクティス

この付録では、特定の機能を実装する際に必要となる可能性のあるその他のトラブルシューティングとベスト プラクティスを示し、説明します。

この付録の内容は、次のとおりです。

Cisco Compatible Extensions バージョン 5 クライアント デバイスのトラブルシューティング

Cisco Compatible Extension クライアントとの通信の問題をトラブルシューティングするために、診断チャネルとクライアント レポーティングの 2 つの機能が設計されています。

（注） これらの機能は、Cisco Compatible Extensions バージョン 5 クライアント デバイスだけでサポートされています。Cisco Compatible Extensions バージョン 5 クライアント デバイス以外での使用、および以前のバージョンが稼働しているクライアントでの使用はサポートされていません。

診断チャネル

診断チャネル機能により、WLAN とのクライアント通信に関する問題のトラブルシューティングが可能になります。困難を抱えたクライアントによって起動する場合も診断チャネルは WLAN です。したがって、クライアントのパスに置かれた通信への障害物が最も少なく、最も堅牢な通信方法が提供されるように設定されます。クライアントが経験した通信の困難の原因を特定する試行において、定義済みの一連のテストをクライアントとアクセス ポイントに受けさせることができます。

（注） コントローラごとに 1 つの WLAN しか診断チャネルを有効にできず、この WLAN のセキュリティはすべて無効となります。

診断チャネルの設定

診断チャネルを設定する手順は、次のとおりです。

ステップ 1　[Configure] > [Controllers] の順に選択します。

ステップ 2　IP アドレスをクリックして、特定のコントローラを選択します。

ステップ 3　左側のサイドバーのメニューから、[WLANs] > [WLAN Configuration] の順に選択します。

ステップ 4　[Select a command] ドロップダウン リストから [Add a WLAN] を選択して新しいプロファイルを作成するか、既存のプロファイル名をクリックします。

（注） 診断テストを実行するための新しい WLAN を作成することを推奨します。

ステップ 5　[WLANs] ページが表示されたら、[Advanced] タブをクリックします。

ステップ 6　この WLAN 上で診断チャネルでのトラブルシューティングを有効にする場合は、[Diagnostic Channel] チェックボックスをオンにします。有効にしない場合は、このチェックボックスをオフのままにします（デフォルト値）。

ステップ 7　[Save] をクリックして変更をコミットします。

WLAN 上の Web 認証セキュリティ

この項では、WLAN に Web 認証セキュリティを実装する場合に役立つトラブルシューティングとベスト プラクティスの手順を説明します。

Web 認証は、WLAN 上のユーザへの Web ベースの認証を可能にするレイヤ 3 のセキュリティ機能です。主にゲスト ネットワークのシナリオで使用されますが、これに限定されるわけではありません。

WLAN が Web 認証セキュリティで設定されると、レイヤ 2 認証（静的 WEP、WPA+PSK、MAC フィルタリングなど）の通過後にログイン ページにリダイレクトされます。ログイン ページは、ローカル デバイスまたは外部 Web サーバに格納され、ロゴ、タイトルなどのカスタマイズが許可されるように変更できます。

WLAN が Web 認証 WLAN で設定された後は、要求された Web サイトに、無線クライアントによって HTTP *get* request が送信されます。コントローラ ファイアウォールによって、指定された URL の DNS 名前解決が許可されます。名前解決後に、コントローラが無線クライアントからの HTTP パケッ

トに割り込んで、ログイン ページにリダイレクトします。クレデンシャルがログイン ページに入力および送信されると、ローカル データベースに対して認証されます。ユーザがローカル データベースで検出されない場合、設定された RADIUS サーバに接触します。

（注）　クライアントと認証エージェント間には、PAP および CHAP 認証が使用されます。RADIUS サーバがこれらのプロトコルをいずれもサポートしていて、Web 認証が許されていることを確認してください。

認証が成功すると、トラフィックを通過させることができます。認証の試行が 3 回失敗すると、クライアントは除外されます。除外されたクライアントは、除外タイムアウト制限を超えるまでアソシエートできません。除外タイムアウト制限は、アグレッシブ ロード バランシングで設定されます。アグレッシブ ロード バランシングは、モバイル クライアントとアソシエートされたアクセス ポイントの間で負荷をアクティブに分散させます。

Web 認証 WLAN は、事前認証の Access Control List（ACL; アクセス コントロール リスト）が設定されることもあります。この ACL は通常の ACL 同様に設定されますが、クライアントが認証に先立って必要とするリソースへのアクセスを許可します。管理者は、インターフェイス セクションを使用して認証後に ACL をクライアントに適用する必要があります。

Web 認証 WLAN は、セッションのタイムアウト値も設定できます。この値によって、クライアントが再度デバイスの認証を行うのに必要な時間が定義されます。値が 0 に設定される場合、これは無限を意味するので、クライアントはログアウト オプションが使用されるまで再度認証されることはありません。http://<VirtualIP>/logout.html でログアウト URL にアクセスできます。

（注）　ログアウト ページを表示するには、クライアント上のポップアップ ブロッカーをすべてオフにします。

Web 認証は、レイヤ 3 セキュリティ下の異なるモードに設定することができます。最も一般的に使用される Web 認証のモードは、次のとおりです。

- Internal Web：http://<virtual IP /DNS name >/login.html を使用した、内部ページへのリダイレクション。カスタマイズ可能です。
- External Web：外部 URL へのリダイレクション。

debug コマンド

次のデバッグ コマンドが使用できます。

```
debug client <client-mac-address>
debug pm ssh-tcp enable
debug pm ssh-appgw enable
debug pm rules enable
debug pm config enable

show client detail <client-mac-address>
debug pem event enable
```

デバッグ戦略

ゲスト トンネリングなしで WLAN に設定された Web 認証に対して、次の戦略を使用します。

ステップ 1　無線 MAC アドレスを使用し書き込むモバイル クライアントを特定します。MS Windows ベースのすべてのシステムでコマンド **prompt > ipconfig /all** を使用します。

ステップ 2　モバイル クライアントの無線を無効にします。

ステップ 3　高速（115200）用シリアル コンソール セットまたはコントローラの管理ポートへの SSH セッション経由で次のデバッグ コマンドを入力にします。

```
debug client <client-mac-address>
debug pm ssh-tcp enable
debug pm ssh-appgw enable
debug pm rules enable
debug pm config enable

show client detail <client-mac-address>

debug pem event enable
debug pem state enable
```

ステップ 4　無線を有効にし、クライアントをアソシエートさせます。クライアントがアソシエートされてから、**show client detail** *client-mac-address* コマンドを入力します。

```
$Router1> show client detail 00:0b:85:09:96:10
Client Username ................................ N/A
AP MAC Address.................................. 00:0b:85:09:96:10
Client State.................................... Associated
Wireless LAN Id................................. 1
BSSID........................................... 00:0b:85:09:96:1f
Channel......................................... 11
IP Address...................................... 10.50.234.3
Association Id.................................. 1
Authentication Algorithm........................ Open System
Reason Code..................................... 0
Status Code..................................... 0
Session Timeout................................. 0
Client CCX version.............................. 3
Mirroring....................................... Disabled
QoS Level....................................... Silver
Diff Serv Code Point (DSCP)..................... disabled
802.1P Priority Tag............................. disabled
WMM Support..................................... Disabled
Mobility State.................................. Local
Internal Mobility State......................... apfMsMmInitial
Mobility Move Count............................. 0
--More-- or (q)uit
Security Policy Completed....................... No
Policy Manager State............................ WEBAUTH_REQD ========**
Policy Manager Rule Created..................... Yes
NPU Fast Fast Notified.......................... Yes
Last Policy Manager State....................... WEBAUTH_REQD
Client Entry Create Time........................ 67733 seconds
Policy Type..................................... N/A
Encryption Cipher............................... None
Management Frame Protection..................... No
EAP Type........................................ Unknown
Interface....................................... management
VLAN............................................ 0
Client Capabilities:
      CF Pollable............................... Not implemented
      CF Poll Request........................... Not implemented
      Short Preamble............................ Implemented
      PBCC...................................... Not implemented
      Channel Agility........................... Not implemented
```

```
      Listen Interval..................................... 0
Client Statistics:
      Number of Bytes Received.................. 188595
      Number of Bytes Sent...................... 19229
      Number of Packets Received................. 3074
--More-- or (q)uit
      Number of Packets Sent..................... 76
      Number of Policy Errors.................... 0
      Radio Signal Strength Indicator............ -41 dBm
      Signal to Noise Ratio...................... 59 dB
Nearby AP Statistics:
        TxExcessiveRetries: 0
        TxRetries: 0
        RtsSuccessCnt: 0
        RtsFailCnt: 0
        TxFiltered: 0
        TxRateProfile: [0,0,0,0,0,0,0,0,0,0,0,0]
      ap:09:96:10(slot 1) ......................
antenna0: 48 seconds ago -45 dBm.................. antenna1: 123 seconds ago -128 dBm
```

ステップ 5　クライアントの PEM 状態が WEBAUTH_REQD であることを確認します。クライアントのブラウザ ページを開いて、次のメッセージを探します。

```
Wed Mar  7 17:59:15 2007: ************* sshpmAddWebRedirectRules: POLICY SEMAPHORE LOCKED
*************
Wed Mar  7 17:59:15 2007: sshpmAddWebRedirectRules: mobile station addr is 10.50.234.3
Wed Mar  7 17:59:15 2007: sshpmAddWebRedirectRules: RuleID for ms 10.50.234.3 is 44
Wed Mar  7 17:59:15 2007: sshpmAddWebRedirectRules: using HTTP-S for web auth (addr:
10.50.234.15).
Wed Mar  7 17:59:15 2007: sshpmAddWebRedirectRules: inbound local http rule created for ms
10.50.234.3 local 1.1.1.1.
Wed Mar  7 17:59:15 2007: sshpmAddWebRedirectRules: inbound http redirect rule created.
Wed Mar  7 17:59:15 2007: sshpmRuleIndexInsert: adding rule for RuleID 44
Wed Mar  7 17:59:15 2007: sshpmRuleIndexInsert: computed raw hash index 02ad3271 for rule
id 0000002c
Wed Mar  7 17:59:15 2007: sshpmRuleIndexInsert: computed adjusted index 00000c32 for rule
id 0000002c
Wed Mar  7 17:59:15 2007: sshpmAddWebRedirectRules: committing rules for ms 10.50.234.3
Wed Mar  7 17:59:15 2007: ************* sshpmPolicyCommitCallback: POLICY SEMAPHORE
UNLOCKED - [unconditionally] *************
Wed Mar  7 17:59:15 2007: sshpmPolicyCommitCallback: called; ContextPtr: 0x2c; Success: 1
Wed Mar  7 17:59:15 2007: ************* sshpmPolicyCommitCallback: POLICY SEMAPHORE
UNLOCKED - [unconditionally] *************
Wed Mar  7 18:02:32 2007: SshPmAppgw/pm_appgw.c:1234/ssh_pm_appgw_request: New application
gateway request for `alg-http@ssh.com': 10.50.234.3.1153 > 10.50.234.1.80 (nat:
10.50.234.1.80) tcp ft=0x00000000 tt=0x00000000
Wed Mar  7 18:02:32 2007: SshPmAppgw/pm_appgw.c:1239/ssh_pm_appgw_request: Packet
attributes: trigger_rule=0x4ecb, tunnel_id=0x0, trd_index=0xddffffff,
prev_trd_index=0xddffffff
Wed Mar  7 18:02:32 2007: SshPmAppgw/pm_appgw.c:1240/ssh_pm_appgw_request: Packet:
Wed Mar  7 18:02:32 2007: 00000000: 4500 0030 0308 4000 8006 0f57 0a32 ea03
E..0..@....W.2..
Wed Mar  7 18:02:32 2007: 00000010: 0a32 ea01 0481 0050 2f42 e3a4 0000 0000
.2.....P/B......
Wed Mar  7 18:02:32 2007: 00000020: 7002 4000 42fe 0000 0204 05b4 0101 0402
p.@.B...........
Wed Mar  7 18:02:32 2007: SshPmStAppgw/pm_st_appgw.c:403/ssh_pm_st_appgw_start: Calling
redirection callback
Wed Mar  7 18:02:32 2007: SshPmAppgw/pm_appgw.c:155/ssh_appgw_redirect: Application
gateway redirect: 10.50.234.1.80 -> 10.50.234.1.80
Wed Mar  7 18:02:32 2007: SshPmStAppgw/pm_st_appgw.c:445/ssh_pm_st_appgw_mappings:
Creating application gateway mappings: 10.50.234.3.1153 > 10.50.234.1.80 (10.50.234.1.80)
Wed Mar  7 18:02:32 2007: SshPmStAppgw/pm_st_appgw.c:102/ssh_pm_appgw_mappings_cb: appgw
connection cached: init flow_index=5967 resp flow_index=5964 event_cnt=718
```

```
Wed Mar  7 18:02:32 2007: SshPmStAppgw/pm_st_appgw.c:493/ssh_pm_st_appgw_mappings_done:
NAT on initiator side
Wed Mar  7 18:02:32 2007:
SshPmStAppgw/pm_st_appgw.c:583/ssh_pm_st_appgw_tcp_responder_stream_done:
ssh_pm_st_appgw_tcp_responder_stream_done: conn->context.responder_stream=0x0
Wed Mar  7 18:02:32 2007:
SshPmStAppgw/pm_st_appgw.c:624/ssh_pm_st_appgw_tcp_responder_stream_done: Opening
initiator stream 10.50.234.1:61611 > 10.76.108.121:2024
Wed Mar  7 18:02:32 2007: SshPmStAppgw/pm_st_appgw.c:154/ssh_pm_appgw_i_flow_enabled:
Initiator flow mode has now been set.
Wed Mar  7 18:02:32 2007: SshPmAppgw/pm_appgw.c:507/ssh_appgw_tcp_listener_callback: New
initiator stream: src=10.50.234.1:61611, dst=10.76.108.121:2024
Wed Mar  7 18:02:32 2007:
SshPmStAppgw/pm_st_appgw.c:646/ssh_pm_st_appgw_tcp_open_initiator_stream: Initiator stream
opened
Wed Mar  7 18:02:32 2007: SshAppgwHttp/appgw_http.c:531/ssh_appgw_http_conn_cb: New TCP
HTTP connection 10.50.234.3.1153 > 10.50.234.1.80
Wed Mar  7 18:02:32 2007: SshAppgwHttp/appgw_http.c:535/ssh_appgw_http_conn_cb: Responder
sees initiator as `10.50.234.15.1153'
Wed Mar  7 18:02:32 2007: SshAppgwHttp/appgw_http.c:539/ssh_appgw_http_conn_cb: Initiator
sees responder as `10.50.234.1.80'
Wed Mar  7 18:02:32 2007: SshAppgwHttp/appgw_http.c:99/ssh_appgw_http_st_wait_input:
entering state st_wait_input: (i) reading_hdr 1 nmsgs 0
Wed Mar  7 18:02:32 2007:
SshAppgwHttpState/appgw_http_state.c:2077/ssh_appgw_http_handle_state: handling: 0 bytes:
Wed Mar  7 18:02:32 2007: SshAppgwHttp/appgw_http.c:136/ssh_appgw_http_st_wait_input: read
-1 bytes (offset 0 data 0)
Wed Mar  7 18:02:32 2007: SshAppgwHttp/appgw_http.c:99/ssh_appgw_http_st_wait_input:
entering state st_wait_input: (r) reading_hdr 1 nmsgs 0
Wed Mar  7 18:02:32 2007:
SshAppgwHttpState/appgw_http_state.c:2077/ssh_appgw_http_handle_state: handling: 0 bytes:
Wed Mar  7 18:02:32 2007: SshAppgwHttp/appgw_http.c:132/ssh_appgw_http_st_wait_input:
appgw_http.c.132: io->src is NULL
Wed Mar  7 18:02:32 2007: SshAppgwHttp/appgw_http.c:136/ssh_appgw_http_st_wait_input: read
-1 bytes (offset 0 data 0)
Wed Mar  7 18:02:32 2007: SshAppgwHttp/appgw_http.c:99/ssh_appgw_http_st_wait_input:
entering state st_wait_input: (i) reading_hdr 1 nmsgs 0
Wed Mar  7 18:02:32 2007:
SshAppgwHttpState/appgw_http_state.c:2077/ssh_appgw_http_handle_state: handling: 0 bytes:
Wed Mar  7 18:02:32 2007: SshAppgwHttp/appgw_http.c:136/ssh_appgw_http_st_wait_input: read
-1 bytes (offset 0 data 0)
Wed Mar  7 18:02:32 2007: SshAppgwHttp/appgw_http.c:99/ssh_appgw_http_st_wait_input:
entering state st_wait_input: (r) reading_hdr 1 nmsgs 0
Wed Mar  7 18:02:32 2007:
SshAppgwHttpState/appgw_http_state.c:2077/ssh_appgw_http_handle_state: handling: 0 bytes:
Wed Mar  7 18:02:36 2007: SshAppgwHttp/appgw_http.c:132/ssh_appgw_http_st_wait_input:
appgw_http.c.132: io->src is NULL
Wed Mar  7 18:02:36 2007: SshAppgwHttp/appgw_http.c:136/ssh_appgw_http_st_wait_input: read
-1 bytes (offset 0 data 0)
Wed Mar  7 18:02:41 2007: SshAppgwHttp/appgw_http.c:99/ssh_appgw_http_st_wait_input:
entering state st_wait_input: (i) reading_hdr 1 nmsgs 0
Wed Mar  7 18:02:41 2007:
SshAppgwHttpState/appgw_http_state.c:2077/ssh_appgw_http_handle_state: handling: 0 bytes:
Wed Mar  7 18:02:41 2007: SshAppgwHttp/appgw_http.c:136/ssh_appgw_http_st_wait_input: read
283 bytes (offset 0 data 0)
Wed Mar  7 18:02:41 2007:
SshAppgwHttpState/appgw_http_state.c:2077/ssh_appgw_http_handle_state: handling: 283
bytes:
Wed Mar  7 18:02:41 2007: 00000000: 4745 5420 2f20 4854 5450 2f31 2e31 0d0a  GET /
HTTP/1.1..
Wed Mar  7 18:02:41 2007: 00000010: 4163 6365 7074 3a20 696d 6167 652f 6769  Accept:
image/gi
Wed Mar  7 18:02:41 2007: 00000020: 662c 2069 6d61 6765 2f78 2d78 6269 746d  f,
image/x-xbitm
```

```
Wed Mar  7 18:02:41 2007: 00000030: 6170 2c20 696d 6167 652f 6a70 6567 2c20  ap,
image/jpeg,
Wed Mar  7 18:02:41 2007: 00000040: 696d 6167 652f 706a 7065 672c 2061 7070  image/pjpeg,
app
Wed Mar  7 18:02:41 2007: 00000050: 6c69 6361 7469 6f6e 2f78 2d73 686f 636b
lication/x-shock
Wed Mar  7 18:02:41 2007: 00000060: 7761 7665 2d66 6c61 7368 2c20 2a2f 2a0d  wave-flash,
*/*.
Wed Mar  7 18:02:41 2007: 00000070: 0a41 6363 6570 742d 4c61 6e67 7561 6765
.Accept-Language
Wed Mar  7 18:02:41 2007: 00000080: 3a20 656e 2d75 730d 0a41 6363 6570 742d  :
en-us..Accept-
Wed Mar  7 18:02:41 2007: 00000090: 456e 636f 6469 6e67 3a20 677a 6970 2c20  Encoding:
gzip,
Wed Mar  7 18:02:41 2007: 000000a0: 6465 666c 6174 650d 0a55 7365 722d 4167
deflate..User-Ag
Wed Mar  7 18:02:41 2007: 000000b0: 656e 743a 204d 6f7a 696c 6c61 2f34 2e30  ent:
Mozilla/4.0
Wed Mar  7 18:02:41 2007: 000000c0: 2028 636f 6d70 6174 6962 6c65 3b20 4d53   (compatible;
MS
Wed Mar  7 18:02:41 2007: 000000d0: 4945 2036 2e30 3b20 5769 6e64 6f77 7320  IE 6.0;
Windows
Wed Mar  7 18:02:41 2007: 000000e0: 4e54 2035 2e31 3b20 5356 3129 0d0a 486f  NT 5.1;
SV1)..Ho
Wed Mar  7 18:02:41 2007: 000000f0: 7374 3a20 3130 2e35 302e 3233 342e 310d  st:
10.50.234.1.
Wed Mar  7 18:02:41 2007: 00000100: 0a43 6f6e 6e65 6374 696f 6e3a 204b 6565  .Connection:
Kee
Wed Mar  7 18:02:41 2007: 00000110: 702d 416c 6976 650d 0a0d 0a              p-Alive....
Wed Mar  7 18:02:41 2007:
SshAppgwHttpState/appgw_http_state.c:985/ssh_appgw_parse_request_line: parsing request
line GET / HTTP/1.1
Wed Mar  7 18:02:41 2007:
SshAppgwHttpState/appgw_http_state.c:1018/ssh_appgw_parse_request_line: internal http
version 3
Wed Mar  7 18:02:41 2007: SshAppgwHttpState/appgw_http_state.c:1155/ssh_appgw_add_method:
caching method 2 for reply 0
Wed Mar  7 18:02:41 2007: SshAppgwHttpState/appgw_http_state.c:1604/ssh_appgw_check_msg:
examining request using service id 34
Wed Mar  7 18:02:41 2007:
SshAppgwHttpState/appgw_http_state.c:594/ssh_appgw_http_get_dst_host: destination host:
10.50.234.1
Wed Mar  7 18:02:41 2007:
SshAppgwHttpState/appgw_http_state.c:1474/ssh_appgw_inject_reply: injecting 404 reply as
msg 0
Wed Mar  7 18:02:41 2007: SshAppgwHttp/appgw_http.c:284/ssh_appgw_http_st_write_data:
entering state st_write_data
Wed Mar  7 18:02:41 2007: SshAppgwHttp/appgw_http.c:99/ssh_appgw_http_st_wait_input:
entering state st_wait_input: (i) reading_hdr 1 nmsgs 1
Wed Mar  7 18:02:41 2007:
SshAppgwHttpState/appgw_http_state.c:2077/ssh_appgw_http_handle_state: handling: 0 bytes:
Wed Mar  7 18:02:41 2007: SshAppgwHttp/appgw_http.c:136/ssh_appgw_http_st_wait_input: read
-1 bytes (offset 0 data 0)
Wed Mar  7 18:02:41 2007: SshAppgwHttp/appgw_http.c:99/ssh_appgw_http_st_wait_input:
entering state st_wait_input: (r) reading_hdr 1 nmsgs 0
Wed Mar  7 18:02:41 2007:
SshAppgwHttpState/appgw_http_state.c:1851/ssh_appgw_http_is_inject: next inject is msg# 0
current msg# 0
Wed Mar  7 18:02:41 2007: SshAppgwHttp/appgw_http.c:207/ssh_appgw_http_st_inject: entering
state st_inject (r): msgs 0
Wed Mar  7 18:02:41 2007: SshAppgwHttp/appgw_http.c:259/ssh_appgw_http_st_inject: closing
connection after inject
Wed Mar  7 18:02:41 2007: SshAppgwHttp/appgw_http.c:400/ssh_appgw_http_st_terminate:
entering state st_terminate (r): teardown 0 terminate i: 1 r: 1
```

```
Wed Mar  7 18:02:45 2007: SshAppgwHttp/appgw_http.c:99/ssh_appgw_http_st_wait_input:
entering state st_wait_input: (i) reading_hdr 1 nmsgs 1
Wed Mar  7 18:02:45 2007:
SshAppgwHttpState/appgw_http_state.c:2077/ssh_appgw_http_handle_state: handling: 0 bytes:
Wed Mar  7 18:02:45 2007: SshAppgwHttp/appgw_http.c:400/ssh_appgw_http_st_terminate:
entering state st_terminate (i): teardown 0 terminate i: 1 r: 1
Wed Mar  7 18:02:45 2007:
SshAppgwHttp/appgw_http.c:732/ssh_appgw_http_connection_terminate: service HTTP-REDIR: TCP
HTTP connection 10.50.234.3.1153 > 10.50.234.1.80 terminated
Wed Mar  7 18:02:45 2007: SshPmStAppgw/pm_st_appgw.c:1094/ssh_pm_st_appgw_terminate:
terminating appgw instance
```

ステップ 6　HTTP GET メッセージが表示されない場合、HTTP パケットがコントローラに到達していません。クライアントがリダイレクションを完了してから、ログインを入力して送信します。

ステップ 7　NPUdevshell hapiMmcDebugScbInfoShow へのクライアントのエントリ（「クライアント MAC アドレス」）を確認します。PEM 状態が WEBAUTH_REQD to RUN から動かない場合、クレデンシャルの問題が存在します。ローカルまたは RADIUS データベース内（設定場所にかかわらず）のクレデンシャルを確認します。

ステップ 8　クライアントに RUN 状態が表示される場合、クライアントからゲートウェイへの確認を行い、トラフィックが通過中かどうか確認します。

RF ヒートマップ分析

シナリオ：アクセス ポイントの矛盾したヒート マップが一部表示されます。アクセス ポイントの一部が、強いヒート マップを示し、他の部分が弱いヒート マップを示します。

分析：これは、一部の隣接アクセス ポイントの RSSI 値を取得し、その他の隣接アクセス ポイントの RSSI 値を取得していない場合に発生することがあります。厚い壁や有線ハウジングがあることが原因で、ヒート マップが正しくなくなる可能性があるため、一方の RSSI 値のみを使用してヒート マップを予測することは推奨されません。

シナリオ：動的ヒート マップを正しく表示できません。

分析：動的ヒート マップを正しく表示できない場合は、次の点を確認します。

- コントローラと Prime Infrastructure の両方でネイバー AP RSSI 値が同じかどうかを確認します。
- 最新の動的ヒート マップ データにヒート マップが更新されるまで、20 分待ちます。
- AP の位置を確認します。

ベスト プラクティス

クライアントがログイン ページにリダイレクトされず、ネットワーク内の DNS 解決を避ける必要がある場合は、**http://***controller-mgmt-ip* と入力します。リダイレクションが発生する場合、問題はネットワーク関連ではありません。

config network web-auth-port *Port* を入力して、コントローラに標準の HTTP ポート (80) 以外のポートを定義します。コントローラは、ポートが割り込みに設定されていても、セキュア HTTP または HTTPS (443) に割り込みません。

RAID カード設定のトラブルシューティング

シナリオ：

偶発的な停電が原因で、NVRAM（不揮発性 RAM）に保存されていた RAID カード設定に関する情報が破損または消去されました。設定情報が失われると、RAID カードは通常モードで起動できません。ただし、RAID カードはハード ドライブに設定情報をバックアップします。RAID カードは、ハード ドライブに保存されているバックアップ設定を認識しましたが、手作業なしでは、その設定情報をデフォルト設定としてをロードしません。

分析：

システムが起動しようとすると、RAID ファームウェアにより、以前の設定に関する情報が失われ、設定ユーティリティをロードするために C キーを押す必要があることを示すエラー メッセージが返されます。エラー メッセージはシリアル コンソールに表示され、入力なしでは起動は進行しません。

次の手順を実行する必要があります。

ステップ 1　シリアル コンソールで、C キーを押して RAID 管理ツールをロードします。RAID ファームウェアにより、外部設定を使用できることが示されます。外部設定はハード ドライブにバックアップされた RAID カード設定です。ただし、RAID ファームウェアはこの設定情報を自動的にロードしません。

ステップ 2　RAID 管理ツールで、次のコマンドを入力します。

-CfgForeign -Import -a0

ステップ 3　サーバをリブートします。

暗号化アクセス用 Cisco.com アカウントの申請

暗号化イメージをダウンロードするには、暗号化アクセス用の Cisco.com アカウントが必要です。

暗号化アクセスを申請する手順は、次のとおりです。

ステップ 1　Cisco.com アカウントを持っている場合は、ステップ 2 に進みます。Cisco.com アカウントを持っていない場合は、次の URL で登録してください。http://tools.cisco.com/RPF/register/register.do

ステップ 2　次の URL にアクセスします。http://tools.cisco.com/legal/k9/controller/do/k9Check.x?eind=Y

[Enter Network Password] ダイアログボックスが表示されます。

ステップ 3　Cisco.com アカウントでログインします。

[Encryption Software Export Distribution Authorization] ページが表示されます。

ステップ 4　リストボックスでソフトウェアを選択して [Submit] をクリックします。

[Encryption Software Export Distribution Authorization] ページが表示されます。

ステップ 5　Encryption Software Export Distribution Authorization Form を検討し、各項目に入力し、[Submit] をクリックします。

「Cisco Encryption Software: Crypto Access Granted」というメッセージが表示されます。

（注）　申請の処理には約 4 時間かかります。この資格付与の処理が完了するまで、ソフトウェアのダウンロードはできません。これに関する通知は送信されません。

ディスクのクリーンアップの実行

Prime Infrastructure のディスク領域が不足していると、アラームがシステムで発生します。また、次のエラーがポップアップ ダイアログボックスに表示されます。

```
The system is running low on diskspace, please refer to online help to perform
disk cleanup.
```

この問題を解決するには、次の CLI コマンドを使用します。

ncs cleanup

このコマンドを使用すると、ディスク領域を解放し、再利用できます。

また、システム ディスクの使用量をモニタすることもできます。詳細については、「システム ディスクの使用量の検査」（P.A-10）を参照してください。

Prime Infrastructure のネットワーク データの収集と保存の管理について詳しくは、次の URL を参照してください。

http://www.cisco.com/en/US/docs/net_mgmt/prime/infrastructure/1.2/user/guide/ManageData.html

システム ディスクの使用量の検査

[Administration] > [Appliance] ページの [Appliance Status] タブを使用して、システム ディスク領域使用率の合計をすぐに確認できます。

[Administration] > [Appliance] > [Appliance Status] の順に選択します。

[Disk Usage] に、現在のストレージ割り当て、および Prime Infrastructure が使用する各メイン ディスク ボリュームの使用率が表示されます。

Prime Infrastructure のパスワードで使用できない特殊文字

パスワードには、「cisco」、「ocsic」、「admin」、または「nimda」は使用できません。また、これらの単語は、大文字と小文字を変えたり、「i」を「1」、「|」、「!」、「o」を「0」、または「s」を「$」に変えても使用できません。

$、'、\、%、&、(、)、;、"、<、>、,、?、| などの特殊文字は、FTP パスワードの一部として使用できません。@、#、^、*、~、_、-、+、=、{、}、[、]、:、.、および / などの特殊文字をパスワードには使用できます。特殊文字「!」（感嘆符）は、パスワード ポリシーが無効の場合に動作します。