O autenticador e o Mensagem-autenticador inválidos do RAIO pesquisam defeitos o guia

Índice

Introdução

Este documento descreve dois mecanismos de segurança do RAIO:

- Encabeçamento do autenticador
- Atributo do Mensagem-autenticador

Este capas de documento o que estes mecanismos de segurança são, como estão usados, e quando você dever esperar a falha da validação.

Encabeçamento do autenticador

Pelo RFC 2865, o encabeçamento do autenticador é 16 bytes por muito tempo. Quando é usado em uma solicitação de acesso, está chamado um autenticador do pedido. Quando é usado em qualquer tipo da resposta, está chamado um autenticador da resposta. É usado para:

- Autenticação da resposta
- Esconder da senha

Autenticação da resposta

Se o server responde com o autenticador correto da resposta, o cliente pode computar se essa resposta foi relacionada a um pedido válido.

O cliente envia o pedido com o encabeçamento aleatório do autenticador. Então, o server que envia a resposta calcula o autenticador da resposta com o uso do pacote de requisição junto com o segredo compartilhado:

```
ResponseAuth = MD5(Code + ID + Length + RequestAuth + Attributes + Secret)
```

O cliente que recebe a resposta executa a mesma operação. Se o resultado é o mesmo, o pacote está correto.

> Nota: O atacante que conhece o valor secreto não pode ao spoof a resposta a menos que puder aspirar o pedido.

Quando deve você esperar a falha da validação?

A falha da validação ocorre se o interruptor não põe em esconderijo o pedido anymore (por exemplo, devido ao intervalo). Você pôde igualmente experimentá-la quando o segredo compartilhado é inválido (sim - a Rejeição de acesso igualmente inclui este encabeçamento). Esta maneira, o dispositivo do acesso de rede (NAD) pode detectar a má combinação secreta compartilhada. É relatada geralmente por clientes/server do Authentication, Authorization, and Accounting (AAA) como uma má combinação de chave compartilhada, mas não revela os detalhes.

Esconder da senha

O encabeçamento do autenticador é usado igualmente a fim evitar enviar o atributo da Senha do usuário no texto simples. O message digest 5 (MD5 - segredo, autenticador) é computado primeiramente. Diversas operações XOR com os pedaços da senha são executadas então. Este método é susceptível para ataques autônomos (tabelas do arco-íris) porque o MD5 não é percebido como um algoritmo de sentido único forte anymore.

Está aqui o script do pitão que computa a Senha do usuário:

```
def Encrypt_Pass(password, authenticator, secret):
       m = md5()
       m.update(secret+authenticator)
       return "".join(chr(ord(x) ^ ord(y)) for x, y in zip(password.ljust
(16,'\0')[:16], m.digest()[:16]))
```

Retransmissões

Se alguns dos atributos na solicitação de acesso do RAIO mudaram (como o RAIO ID, username, e assim por diante), o campo novo do autenticador deve ser gerado e todos campos restantes que dependem dele devem ser recalculados. Se esta é uma retransmissão, nada deve mudar.

Relatório

O significado do encabeçamento do autenticador é diferente para uma solicitação de acesso e um Contabilidade-pedido.

Para uma solicitação de acesso, o autenticador é gerado aleatoriamente e espera-se receber uma resposta com o ResponseAuthenticator calculado corretamente, que mostra que a resposta esteve relacionada a esse pedido específico.

Para um Contabilidade-pedido, o autenticador não é aleatório, mas calcula-se (conforme o RFC 2866):

```
RequestAuth = MD5(Code + ID + Length + 16 zero octets + Attributes + Secret)
```

Esta maneira, o server pode verificar a mensagem da contabilidade imediatamente e deixar cair o pacote se o valor voltado a calcular não combina o valor do autenticador. Os retornos do Identity Services Engine (ISE):

```
11038 RADIUS Accounting-Request header contains invalid Authenticator field
```

O motivo típico para este é a chave secreta compartilhada incorreta.

Atributo do Mensagem-autenticador

O atributo do Mensagem-autenticador é o atributo RADIUS definido no RFC 3579. É usado para uma finalidade similar: para assinar e validar. Mas esta vez, não é usado a fim validar uma resposta mas um pedido.

O cliente que envia uma solicitação de acesso (ele pode igualmente ser um server que responda com um Acesso-desafio) computa o Hash-Based Message Authentication Code (HMAC)-MD5 de seu próprio pacote, e adiciona então o atributo do Mensagem-autenticador como uma assinatura. Então, o server pode verificar que executa a mesma operação.

A fórmula olha similar ao encabeçamento do autenticador:

```
Message-Authenticator = HMAC-MD5 (Type, Identifier, Length, Request Authenticator,
Attributes)
```

A função HMAC-MD5 recolhe dois argumentos:

- O payload do pacote, que inclui o campo de um Mensagem-autenticador de 16 bytes encheu-se com os zero
- O segredo compartilhado

Quando deve o Mensagem-autenticador ser usado?

O Mensagem-autenticador DEVE ser usado para cada pacote, que inclui a mensagem do Extensible Authentication Protocol (EAP) (RFC 3579). Isto inclui o cliente que envia a solicitação de acesso e o server que responde com o Acesso-desafio. O outro lado deve silenciosamente deixar cair o pacote se a validação falha.

Quando deve você esperar a falha da validação?

A falha da validação ocorrerá quando o segredo compartilhado é inválido. Então, o servidor AAA não pode validar o pedido.

Os relatórios ISE:

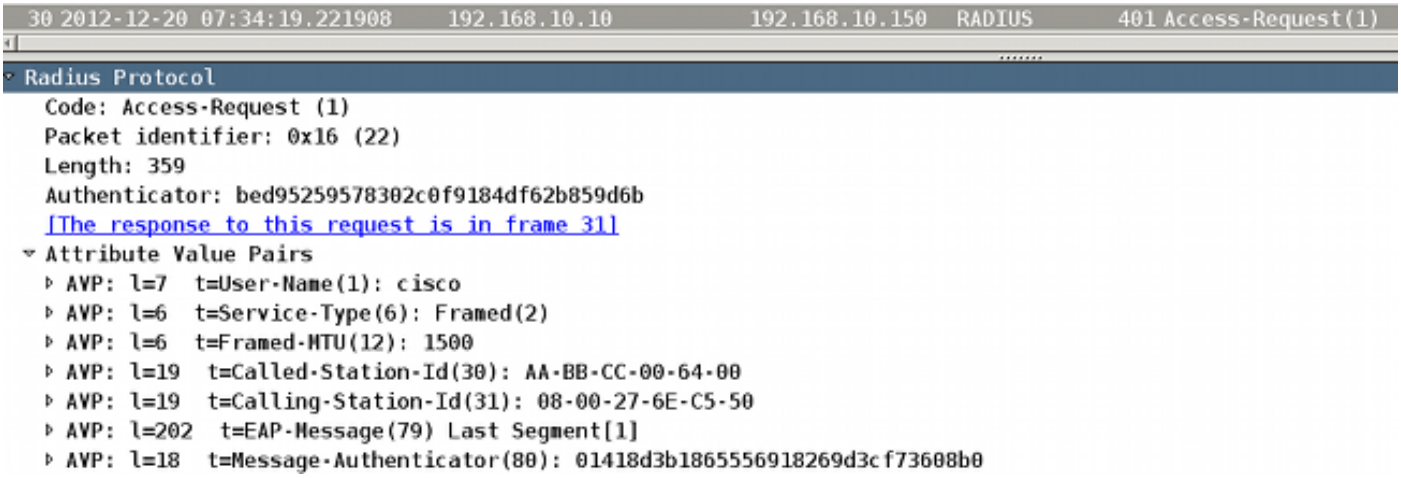
`11036 The Message-Authenticator Radius Attribute is invalid.`

Isto ocorre geralmente no estágio posterior quando o mensagem EAP é anexado. O primeiro pacote de informação de RADIUS da sessão do 802.1x não inclui o mensagem EAP; não há nenhum campo do Mensagem-autenticador e não é possível verificar o pedido, mas nessa fase, o cliente pode validar a resposta com o uso do campo do autenticador.

Valide o atributo do Mensagem-autenticador

Está aqui um exemplo para ilustrar como você conta manualmente o valor a fim se certificar que está computado corretamente.

O pacote número 30 (solicitação de acesso) foi escolhido. É no meio da sessão EAP, e o pacote inclui o campo do Mensagem-autenticador. O alvo é verificar que o Mensagem-autenticador está correto:

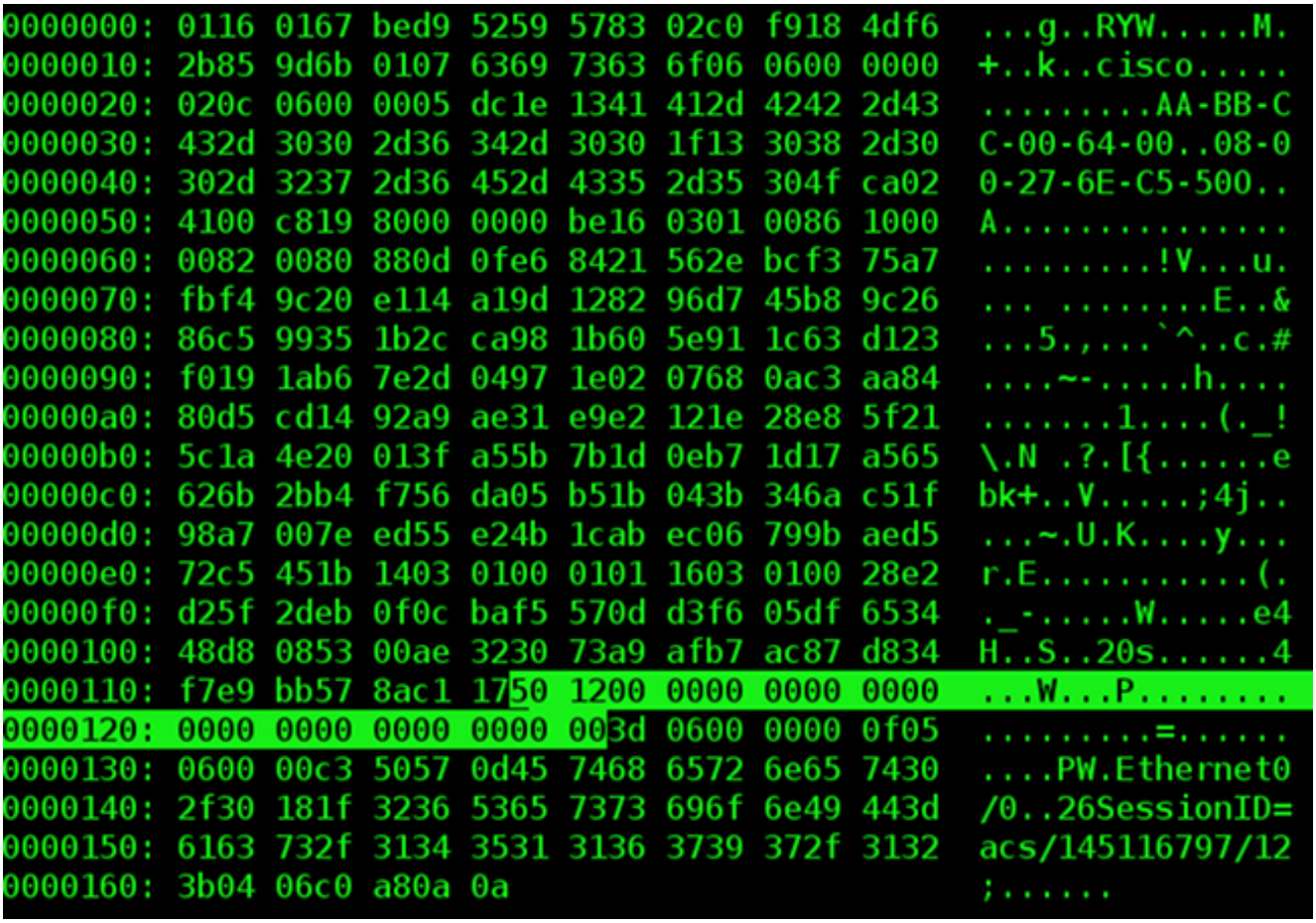


1. Clicar com o botão direito o **protocolo de raio** e escolha **bytes de pacote selecionados exportação**.

2. Escreva esse payload do RAIO a um arquivo (dados binários).

3. A fim computar o campo do Mensagem-autenticador, você deve pôr zero lá e computar o HMAC-MD5.

   Por exemplo, quando você se usa encantar/editor binário, tal como o vim, depois que você datilografa “: %! o xxd”, que comuta para encantar o modo e os zero 16 bytes que começam depois que "5012" (50hex é 80 em dezembro que é tipo do Mensagem-autenticador, e 12 é o tamanho que é 18 que incluem o encabeçamento dos pares de valor de atributo (AVP)):



Em seguida essa alteração, o payload está pronta. É necessário retornar de volta a encanta/modo binário (tipo: “: %! o xxd - r”) e salvar o arquivo (“: wq”).

4. Use o OpenSSL a fim computar o HMAC-MD5:

```
pluton # cat packet30-clear-msgauth.bin | openssl dgst -md5 -hmac 'cisco'
(stdin)= 01418d3b1865556918269d3cf73608b0
```

A função HMAD-MD5 toma dois argumentos: primeiro da entrada padrão (stdin) é a mensagem própria e segunda é o segredo compartilhado (Cisco neste exemplo). O resultado é exatamente o mesmo valor como o Mensagem-autenticador anexado ao pacote de solicitação de acesso do RAIO.

O mesmos podem ser computados com o uso do script do pitão:

```
pluton # cat hmac.py
#!/usr/bin/env python

import base64
import hmac
import hashlib

f = open('packet30-clear-msgauth.bin', 'rb')
try:
   body = f.read()
finally:
   f.close()

digest = hmac.new('cisco', body, hashlib.md5)
d=digest.hexdigest()
print d

pluton # python hmac.py
```

```
01418d3b1865556918269d3cf73608b0
```

O exemplo anterior apresenta como calcular o campo do Mensagem-autenticador da solicitação de acesso. Para o Acesso-desafio, a aceitação de acesso, e a Rejeição de acesso, a lógica é exatamente a mesma, mas é importante recordar que o autenticador do pedido deve ser usado, que é fornecido no pacote de solicitação de acesso precedente.

Informações Relacionadas

- RFC 2865
- RFC 2866
- RFC 3579
- Suporte Técnico e Documentação - Cisco Systems