



## Nota de aplicação

### Chaves exclusivas para ATSHA204

#### Características

- Uso do número de série exclusivo Atmel® ATSHA204 e uma chave raiz para criar um Chave Única (Chave Diversificada)
- Configurando o ATSHA204 com Chaves Únicas
- Autenticando a Chave Única usando um Host ATSHA204 contendo a Chave Raiz
- Descrição da Calculadora de Chave Diversificada em ACES (Atmel Crypto Evaluation Estúdio)
- Demonstração de validação de Host usando o comando DeriveKey
- Demonstração de validação de Host usando o comando GenDig
- Pseudo Código para validação de Host — para sistemas que não possuem um Host ATSHA204

#### Descrição

Uma chave exclusiva pode ser criada para cada cliente com base em seu número de série e uma chave raiz. Isso é conhecido como diversificação de chaves. Como cada dispositivo cliente é programado com um segredo exclusivo, a chave diversificada tem menos valor para um invasor.

Este passo a passo configurará o dispositivo ATSHA204 com uma chave diversificada com base na combinação criptográfica de uma chave raiz com o número de série ATSHA204, que é garantido como exclusivo. Depois de configurar a Chave Diversificada, este passo a passo continuará com um passo a passo para gravar essa Chave Diversificada no dispositivo Cliente.

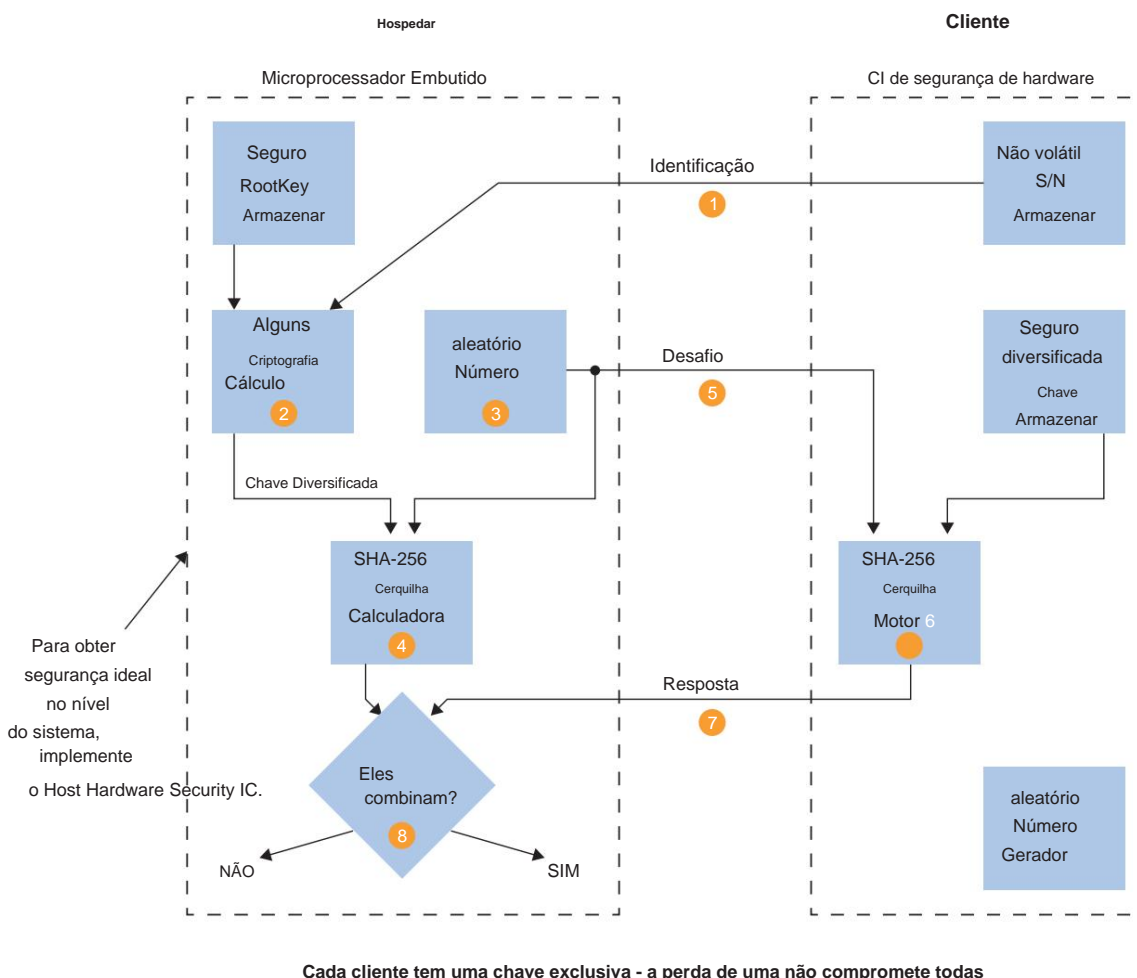
Depois que o cliente é configurado, uma explicação de como um sistema pode validar a chave configurada executando um MAC no cliente diversificado e comparando o resumo resultante com o resumo gerado por um cálculo criptográfico equivalente usando o número de série do cliente e a chave raiz.

Uma demonstração de como o comando GenDig ou o comando DeriveKey pode ser usado por um dispositivo host ATSHA204 para validar a chave diversificada do cliente ATSHA204 também será resumida.

# 1. Descrição Chave Diversificada

Conforme mostrado na [Figura 1-1](#), o Host autentica uma Chave Diversificada do Cliente usando a Chave Raiz que foi usada para calcular a Chave Diversificada do Cliente. O cálculo da chave diversificada combina criptograficamente o número de série do cliente com a chave raiz armazenada no host. Como as Chaves Diversificadas são baseadas em uma Chave Raiz, o Host só precisa saber o Número de Série do Cliente para validar a Chave Diversificada do Cliente.

**Figura 1-1. O host autentica uma chave diversificada de cliente usando a chave raiz**



## 2. Passos passo a passo

As etapas nesta seção descrevem o processo de configuração e autenticação de chaves diversificadas.

### 2.1 Configuração do Dispositivo

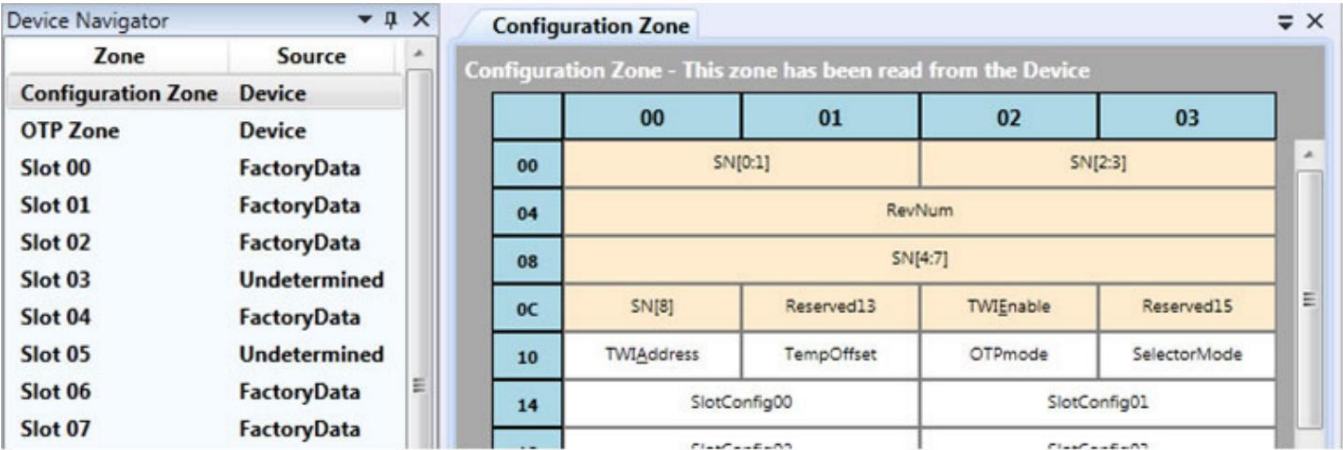
Para este passo a passo, comece configurando a zona de configuração no dispositivo ATSHA204. Esta configuração atuará como Host e Cliente ATSHA204. Essa configuração usa um único dispositivo para demonstrar os conceitos; em um sistema real, o dispositivo host seria separado. [A Tabela 2-1](#) fornece os bytes de descrição e configuração para cada slot usado.

Tabela 2-1. Configurações de slot

Título do espaço		Descrição	Configuração do Slot
00	Cliente Chave Diversificada	<b>Slot do Cliente:</b> Este slot será diversificado usando o Número de Série e a Chave Raiz do Host.	Leia – É Segredo Escreva – Nunca Bytes – 8F 8F
01	Destino do host	<b>Slot do Host:</b> Este é o slot de destino definido para o comando DeriveKey.	Read – É secreto, CheckOnly Write – DeriveKey (pai 2) Bytes - 9F 32
02	Chave Raiz do Host	<b>Raiz usada para diversificação de chaves:</b> Use o Comando DeriveKey para verificar a Chave Diversificada do Cliente. Esta chave deve ser programada no Host ATSHA204.	Leia – É Segredo Escreva – Nunca Bytes – 8F 8F
03	Chave Raiz do Host	<b>Raiz usada para diversificação de chaves:</b> Use o Comando GenDig para verificar a Chave Diversificada do Cliente. Esta chave deve ser programada no Host ATSHA204.	Read – É secreto, CheckOnly Escreva – Nunca Bytes - 9F 8F

1. Inicie o ACES Configuration Environment (CE) com um dispositivo ATSHA204 *desbloqueado* (use um AT88CK101 ou um kit de desenvolvimento AT88CK454).
2. Selecione **Zona de configuração** no **Navegador de dispositivo**, conforme mostrado na [Figura 2-1](#).

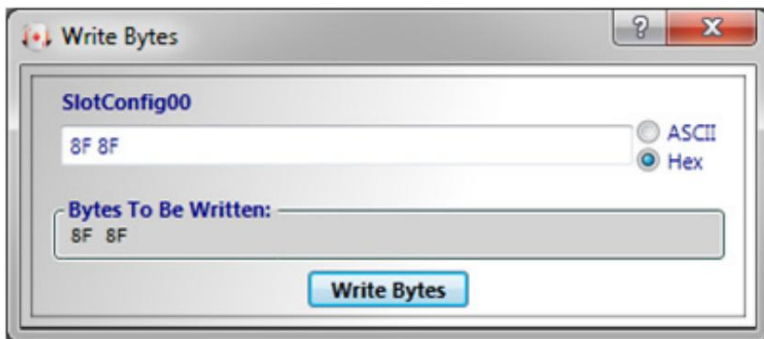
Figura 2-1. Selecione a zona de configuração



3. Clique no local de memória **SlotConfig00** no mapa de memória.

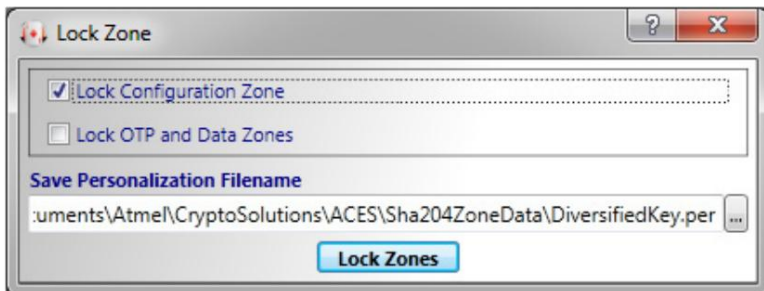
4. A caixa de diálogo **Write Bytes** será exibida conforme mostrado na [Figura 2-2](#).

Figura 2-2. Caixa de Diálogo Write Bytes — SlotConfig00



5. Digite a configuração do Slot 00 no campo **SlotConfig00** da [Tabela 2-1](#) (**8F 8F**). • Repita para Slot 01 (**9F 32**). • Repita para o Slot 02 (**8F 8F**). • Repita para o Slot 03 (**9F 8F**).
6. Bloqueie a zona de configuração.
  - Selecione **Ferramentas > Bloquear zonas** no menu.
  - A caixa de diálogo **Lock Zone** será exibida conforme mostrado na [Figura 2-3](#). • Marque a caixa de seleção **Bloquear zona de configuração** e clique no botão **Bloquear zonas**. • A mensagem **Bloqueio bem-sucedido** será exibida.

Figura 2-3. Caixa de Diálogo de Zona de Bloqueio



7. Inicie a caixa de diálogo **Cálculo de chaves diversificadas**.

- Selecione **Ferramentas > Calcular Chaves Diversificadas** no menu.

A caixa de diálogo **Cálculo de chave diversificada** será exibida conforme mostrado na [Figura 2-4](#).

Observação: esta caixa de diálogo atualiza dinamicamente a chave diversificada calculada à medida que as entradas são modificadas.

- O cálculo usado para esta caixa de diálogo é definido pelo comando DeriveKey.

Figura 2-4. Caixa de Diálogo de Cálculo de Chave Diversificada

8. Configure as **Entradas Chave Diversificadas** de acordo com a configuração mostrada na

[Tabela 2-1](#). • Defina o **slot de destino**

**do host** como 1. • Defina o **valor da chave raiz** como três (use um segredo exclusivo aqui, se tiver um). • O **número de série do dispositivo** será lido do dispositivo e pré-carregado.

- Defina o **teclado numérico de série** para sete (qualquer teclado serve. Normalmente, todos os zeros).

9. Os **Bytes de entrada** referem-se aos bytes que serão passados para o mecanismo Atmel ATSHA256.

- Os bytes e a ordem dos bytes são definidos no comando GenDig.
- A

TempKey é o SN + SnPad que pode ser inicializado com o comando Nonce.

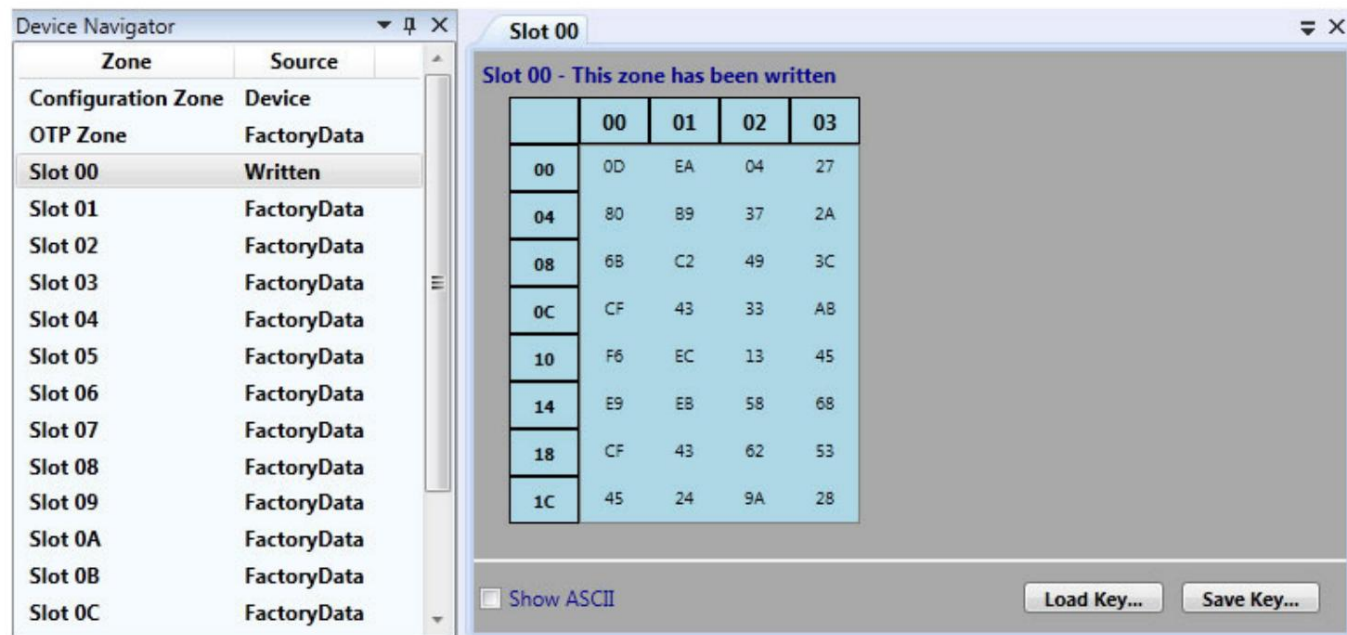
10. A Chave Diversificada calculada é o resultado que deve ser gravado na Chave Diversificada do Cliente (Slot 00).

Observação: esse cálculo combina criptograficamente a chave raiz e o número de série do dispositivo.

- Deixe a caixa de diálogo **Cálculo de chave diversificada** aberta para uso posterior.

11. Selecione **Slot 00** no **Device Navigator**, conforme mostrado na [Figura 2-5](#).

**Figura 2-5. Slot 00 mostrando dados-chave diversificados**



12. Configuração do Cliente — Grave a Chave Diversificada calculada no Slot 00 do ATSHA204.

- Clique três vezes nos dados Calculated Diversified Key na caixa de diálogo **Calculated Diversified Key** para selecionar todos os dados.
- Copie os dados para a área de transferência.
- Clique em qualquer local na zona de memória. A caixa de diálogo **Write Zone** será exibida conforme mostrado na [Figura 2-6](#).
- Cole os dados Chave Diversificada no campo **Dados a Gravar**.
- Clique no botão **Write To Zone**.

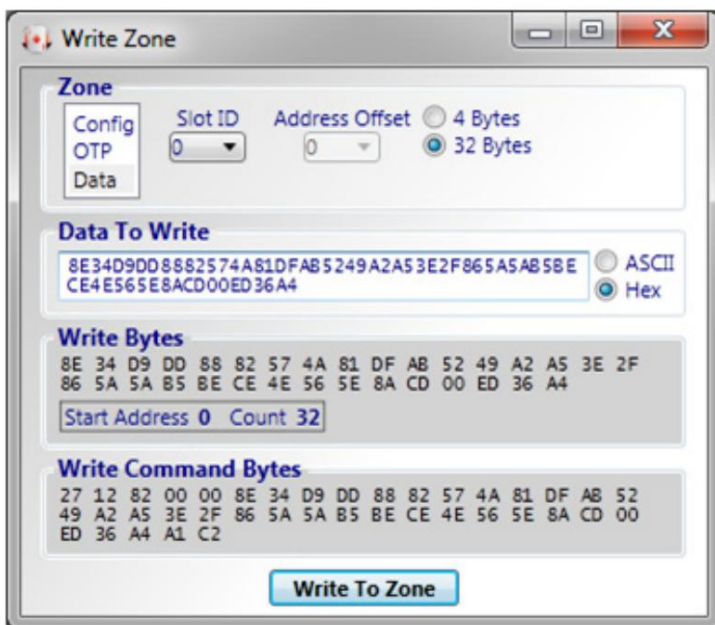
13. Configuração do host — Grave a chave raiz no slot 02 e no slot 03 do ATSHA204. Siga estas etapas para escrever a chave raiz (todos os três ou chave única) que foi usada para gerar a chave diversificada.

• Clique em qualquer local na zona de Memória do Slot 02. A caixa de diálogo **Write Zone** será exibida conforme mostrado na

[Figura 2-6](#).

- Cole os dados da chave raiz (todos os três ou chave exclusiva) no campo **Dados a serem gravados**.
- Clique no botão **Write To Zone**.
- Repita essas etapas de Gravação para o Slot 03.

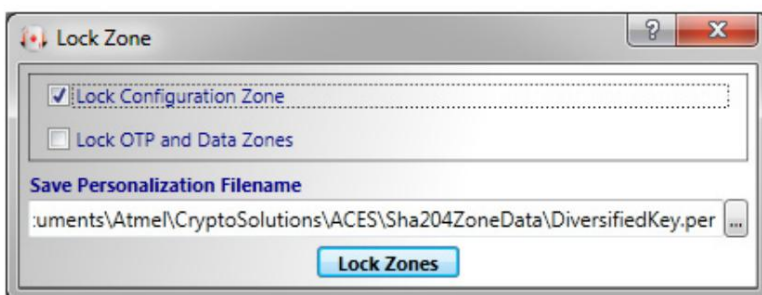
Figura 2-6. Caixa de Diálogo Write Zone — Write Slot 00



14. Bloqueie as zonas OTP e Data.

- Selecione o menu **Ferramentas > Bloquear zonas**.
- A caixa de diálogo **Lock Zone** será exibida conforme mostrado na Figura 2-7. • Selecione a caixa de seleção **Lock OTP and Data Zones** e clique no botão **Lock Zones**.
- A mensagem **Bloqueio bem-sucedido** será exibida.

Figura 2-7. Caixa de Diálogo de Cálculo de Chave Diversificada





## 2.2 Validando a Chave Diversificada

A Chave Diversificada agora foi configurada no Cliente (Slot 00).

Nota: A Chave Diversificada *usa* a Chave Raiz no cálculo criptográfico que a gerou — o Cliente não precisa *ter* o RootKey programado nele.

$$\text{DiversifiedKey} = \text{SHA256}(\text{RootKey}, \text{SerialNumber}, \dots)$$

Além disso, quando o Host tem conhecimento da RootKey, apenas o SerialNumber precisa estar disponível para gerar a DiversifiedKey. Como o SerialNumber pode ser lido de cada Cliente ATSHA204, o Host pode validar a Chave Diversificada de várias maneiras diferentes:

• Usando o comando DeriveKey em um ATSHA204 programado com a Root Key (por exemplo, Slot 02). • Utilizando o comando GenDig em um ATSHA204 programado com a Root Key (ex. Slot 03). • Usando o código do sistema que tem acesso à chave raiz. Para a maioria dos sistemas, esta técnica *não* é recomendada.

Cada uma dessas validações da Chave Diversificada será demonstrada.

### 2.2.1 Pseudocódigo de Validação

A primeira técnica de validação que será examinada é o Pseudo Code Host. Esta técnica *não* é recomendada, pois na maioria dos sistemas, a chave raiz deve ser usada em claro e não pode ser armazenada com segurança no firmware. Esta seção é útil para microprocessadores seguros e para ilustrar os cálculos que são executados internamente no ATSHA204.

---

#### Pseudocódigo de validação de chave diversificada — Código do sistema com RootKey

---

```
// Inicializa a comunicação sha204p_init();

// Definir o dispositivo cliente
sha204p_set_device_id(CLIENT_ID);

// Ativa o ATSHA204 sha204c_wakeup();

// Protótipo da Função: resultBuf = sha204m_execute(command, param1, param2, data)

// Lê os primeiros 32 bytes da zona de configuração para obter o número de série do cliente snRead =
sha204m_execute(SHA204_READ, 0x80, 0x00, 0x00);

// Analisa o SerialNumber do cliente serialNumber =
snRead[0:3] + snRead[8:12];

// Gera um número aleatório no Host para o desafio de 32 bytes randChal =
sha204m_execute(SHA204_RANDOM, 0x00, 0x0000, null);

// Execute um comando MAC no ATSHA204 e salve o resumo param1Mac = 0x00; param2Mac
= [00, 00]; deviceDigest =
sha204m_execute(SHA204_MAC,
param1Mac, param2Mac, randChal);
```



```

// Calcula a chave diversificada usando o cálculo DeriveKey e um soft SHA 256

rootKey =      ... // Segredo de 32 bytes aqui
opCodeDk = 0x1C;
param1 = 0x04;
param2 = ... // ID do slot de 2 bytes aqui (ordem de byte LSB 0x0X 00) sn8 = ... // 1 byte SN[8] aqui

sn01 =      ... // 2 bytes SN[0:1] aqui
zeros =      ... // 25 bytes de 0's aqui ... // 23 bytes de
snPad =      pad aqui divKey =

sha256(rootKey+opCode+param1+param2+sn8+sn01+zeros+serialNumber+snPad);

// Executa um MAC na chave diversificada calculada // usando o cálculo do
comando ATSHA204 MAC e um soft SHA-256 opCodeMac = 0x08; otpZeros = [00, 00, 00, 00, 00, 00, 00,
00, 00, 00, 00, 00, 00]; //
13 bytes de
zeros
sn23 = [00, 00]; // 2 bytes SN[2:3], usa zeros sn47 = [00, 00, 00, 00]; // 4
bytes SN[4:7], use zeros macBytes =

divKey+randChal+opCodeMac+param1Mac+param2Mac+otpZeros+sn8+sn47+sn01+sn23; softDigest = sha256(macBytes);

// Compara os resumos resultantes do ATSHA204 e do soft MAC match = deviceDigest == softDigest;

```

### 2.2.2 Leia o número de série do cliente e execute o comando MAC

Os próximos dois métodos envolvem o uso do ACES com a [Etapa 1.](#); leia o SerialNumber e o [Passo 2.](#); execute o Comando MAC no slot de Chave Diversificada.

1. Execute Read — Leia o número de série

- Selecione o menu **Ferramentas > Construtor de comandos**.
- A caixa de diálogo **Command Builder** será exibida conforme mostrado na [Figura 2-8](#). • Na lista suspensa **OpCode**, selecione o comando **Ler**. • Defina a **zona** para **80** (= 00 e 80) que indica 32 bytes lidos da zona de configuração.
- Defina o **endereço** como **0000**.
- Clique no botão **Executar Comando**.
- O campo **Pacote de Resposta** conterà os bytes que foram lidos.

2. Isole o SerialNumber.

- O número de série de nove bytes são os bytes [0:3] e [8:12]. • Para este exemplo: 0123375205975AEEEE.

Figura 2-8. Ler SerialNumber — Construtor de Comandos

The screenshot shows the 'Command Builder' dialog box with the following configuration:

- Send Count:** 07
- Response Count:** 23
- Command Packet:**
  - OpCode:** Read
  - Zone:** 80
  - Address:** 0000
  - Data:** (empty)
- Send Details:**
  - Send Count:** 07
  - Send Packet:** 02 80 00 00
  - Send Checksum:** 09 AD
- Response Details:**
  - Response Count:** 23
  - Response Packet:** 01 23 37 52 00 04 05 00 05 97 5A EE EE 55 00 FF C8 00 55 00 8F 8F 9F 32 8F 8F 9F 8F 94 40 A0 85
  - Response Checksum:** 91 C3

At the bottom of the dialog is an 'Execute Command' button.

3. Execute MAC — Obtenha o Digest para o slot Diversified Key.
- Deixe a caixa de diálogo **Construtor de comandos** aberta. •
- Na lista suspensa **OpCode**, selecione o comando **MAC**.
- Defina o **Modo** como **00**.
  - Defina o **KeyID** como **0000**. •
- Defina os **Dados** para o desafio de entrada (todos aqui).
- Clique no botão **Executar Comando**.
  - O campo **Pacote de resposta** conterá o resumo.

**Figura 2-9. MAC — Construtor de comandos**

[illegible]

## 2.3 Validar usando o comando GenDig

Para validar o Cliente, siga os seguintes passos utilizando o Comando GenDig. Esta sequência representa a sequência do Host que será realizada para validar o Cliente.

1. Execute **Nonce** — Inicialize TempKey com SerialNumber + SnPad.
  - Selecione o menu **Ferramentas > Construtor de comandos**.
  - A caixa de diálogo **Command Builder** será exibida conforme mostrado na [Figura 2-10](#). • Na lista suspensa **OpCode**, selecione o comando **Nonce**. • Defina o **modo** como **03**, que indica o modo de passagem.
  - Defina os **dados** como SerialNumber + SnPad.
  - Clique no botão **Executar Comando**.
  - O campo **Pacote de resposta** conterá **00**, indicando sucesso.

**Figura 2-10. Nonce — Construtor de comandos**

[illegible]

## 2. Execute GenDig — Inicialize TempKey com a Chave Diversificada.

- Deixe a caixa de diálogo **Construtor de comandos** aberta.

Na lista suspensa **OpCode**, selecione o comando **GenDig**.

- Defina **MemZone** como **02**, que indica a zona de dados.
- Defina **KeyID** como **0300** (LSB). Este é o slot do Host configurado para validação GenDig da Chave Diversificada.
- Defina os **dados** como **1C040100**. Este é o **OtherData** para GenDig que torna o cálculo de criptografia o mesmo que DeriveKey.
- Clique no botão **Executar Comando**.
- O campo **Pacote de resposta** conterá **00**, indicando sucesso.

Figura 2-11. GenDig — Construtor de comandos

The screenshot shows the 'Command Builder' window with the following configuration:

Command Packet	
OpCode:	GenDig
MemZone	02
KeyID	0300
Data:	1C040100

Send Details	
Send Count:	08
Send Packet:	15 02 03 00 1C 04 01 00
Send Checksum:	8C 6B

Response Details	
Response Count:	04
Response Packet:	00
Response Checksum:	03 40

At the bottom of the window is an 'Execute Command' button.

Atmel

- **Desafio** = Todos. **Resposta**  
= Resumo do resultado do comando MAC do cliente. **OtherData** = 08 (MAC  
OpCode) + 00 00 00 00 00 00 00 00 00 00 00 00 (12 bytes de 00).  
e no botão **Executar Comando**.

Command Builder	
Send Count: 54    Response Count: 04	
<b>Command Packet</b>	
<b>OpCode:</b>	CheckMac
<b>Mode</b>	06
<b>KeyID</b>	0100
<b>Data:</b>	111 11111111111111E205CECE79C28AAF25E8491974509188B4CC D0E68FE5015DE94D968E1E5621D508000000000000000000 0000
<b>Send Details</b>	
<b>Send Count:</b>	54
<b>Send Packet:</b>	28 06 01 00 11 E2 05 CE CE 79 C2 8A AF 25 E8 49 19 74 50 91 88 B4 CC D0 E6 8F E5 01 5D E9 4D 96 BE 1E 56 21 D5 08 00 00 00 00 00 00 00 00 00 00 00 00 00 00
<b>Send Checksum:</b>	12 99
<b>Response Details</b>	
<b>Response Count:</b>	04
<b>Response Packet:</b>	00
<b>Response Checksum:</b>	03 40
<b>Execute Command</b>	

## 2.4 Validar usando o comando DeriveKey 1.

Execute Nonce — Initialize TempKey com SerialNumber + SnPad.

- Selecione o menu **Ferramentas > Construtor de comandos**.
- A caixa de diálogo **Command Builder** será exibida conforme mostrado na [Figura 2-13](#). • Na lista suspensa **OpCode**, selecione o comando **None**. • Defina o **modo** como **03**, que indica o modo de passagem.
- Defina os **dados** como SerialNumber + SnPad.
- Clique no botão **Executar Comando**.
- O campo **Pacote de resposta** conterá **00**, indicando sucesso.

**Figura 2-13. Nonce — Construtor de comandos**

[illegible]



2. Execute DeriveKey — Grave a chave diversificada do cliente em um slot no host. • Na lista suspensa

**OpCode**, selecione o comando **DeriveKey**. • Defina **Random** como **04**. Isso

corresponde ao sinalizador de origem TempKey do modo de passagem. • Defina **TargetKey** como **0100** (LSB).

Este slot de Host está configurado para um destino DeriveKey.

• Clique no botão **Executar Comando**.

• O campo **Pacote de resposta** conterá **00**, indicando sucesso.

Figura 2-14. DeriveKey — Construtor de comandos

The screenshot shows the 'Command Builder' window with the following configuration:

Command Packet	
OpCode:	DeriveKey
Random	04
TargetKey	0100
Data:	

Send Details	
Send Count:	07
Send Packet:	1C 04 01 00
Send Checksum:	80 4F

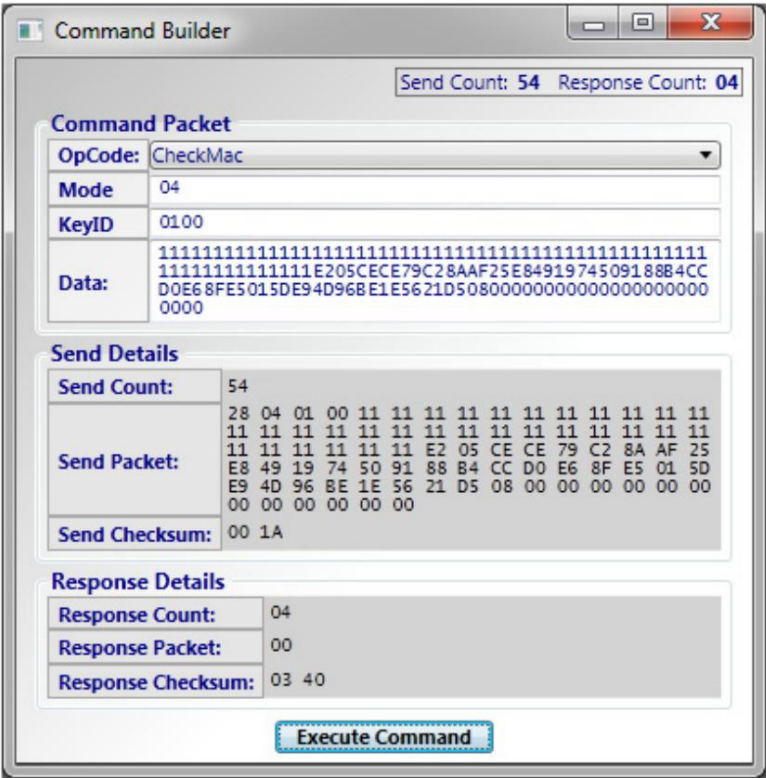
  

Response Details	
Response Count:	04
Response Packet:	00
Response Checksum:	03 40

At the bottom of the window is a button labeled 'Execute Command'.

3. Execute CheckMac — Compare Client Digest com o MAC da Diversified Key derivada (agora no Slot 01).
- Deixe a caixa de diálogo **Construtor de comandos** aberta.
- Na lista suspensa **OpCode**, selecione o comando **CheckMac**.
- Defina o **Modo** como **06** (= 04 e 02). Use TempKey e corresponda ao sinalizador de origem TempKey.
  - Defina **KeyID** como **0100**. Esse valor é ignorado pelo CheckMac ao usar TempKey.
  - Defina os **Dados** como Desafio + Resposta + OutrosDados.
- Desafio = Todos.   • Resposta
- = Resumo do resultado do comando MAC do cliente.   • OtherData = 08 (MAC
- OpCode) + 00 00 00 00 00 00 00 00 00 00 00 00 (12 bytes de 00).
- Clique no botão **Executar Comando**.
  - O campo **Pacote de resposta** conterá **00**, indicando que os resumos correspondem.

Figura 2-15. CheckMac — Construtor de comandos



3. Histórico de revisões

documento não	Data	Comentários
8841A	04/2013	Liberação inicial do documento.



**Atmel Corporation** 1600 Technology Drive, San Jose, CA 95110 EUA **T:** (+1)(408) 441.0311 **F:** (+1)(408) 436.4200 | **www.atmel.com**

© 2013 Atmel Corporation. Todos os direitos reservados. / Rev.: Atmel-8841A-CryptoAuth-ATSHA204-Unique-Keys-ApplicationNote\_042013

Atmel®, o logotipo da Atmel e suas combinações, Enabling Unlimited Possibilities®, CryptoAuthentication™ e outros são marcas registradas ou marcas comerciais da Atmel Corporation ou de suas subsidiárias. Outros termos e nomes de produtos podem ser marcas comerciais de terceiros.

**ISENÇÃO DE RESPONSABILIDADE:** As informações neste documento são fornecidas em relação aos produtos Atmel. Nenhuma licença, expressa ou implícita, por preclusão ou de outra forma, para qualquer direito de propriedade intelectual é concedida por este documento ou em conexão com a venda de produtos Atmel. EXCETO CONFORME ESTABELECIDO NOS TERMOS E CONDIÇÕES DE VENDAS DA ATMEL LOCALIZADOS NO SITE DA ATMEL, A ATMEL NÃO ASSUME NENHUMA RESPONSABILIDADE E RENUNCIA A QUALQUER GARANTIA EXPRESSA, IMPLÍCITA OU LEGAL RELACIONADA A SEUS PRODUTOS, INCLUINDO, SEM LIMITAÇÃO, A GARANTIA IMPLÍCITA DE COMERCIALIZABILIDADE, ADEQUAÇÃO PARA UMA FINALIDADE ESPECÍFICA OU NÃO VIOLAÇÃO. EM NENHUM CASO A ATMEL SERÁ RESPONSÁVEL POR QUAISQUER DANOS DIRETOS, INDIRETOS, CONSEQUENTES, PUNITIVOS, ESPECIAIS OU INCIDENTAIS (INCLUINDO, SEM LIMITAÇÃO, DANOS POR PERDAS E LUCROS, INTERRUPÇÃO DE NEGÓCIOS OU PERDA DE INFORMAÇÕES) DECORRENTES DO USO OU INCAPACIDADE DE USO ESTE DOCUMENTO, MESMO QUE A ATMEL TENHA SIDO AVISADA DA POSSIBILIDADE DE TAIS DANOS. A Atmel não faz representações ou garantias com relação à precisão ou integridade do conteúdo deste documento e reserva-se o direito de fazer alterações nas especificações e descrições de produtos a qualquer momento sem aviso prévio. A Atmel não se compromete a atualizar as informações aqui contidas. Salvo disposição em contrário, os produtos Atmel não são adequados e não devem ser usados em aplicações automotivas. Os produtos da Atmel não são destinados, autorizados ou garantidos para uso como componentes em aplicações destinadas a dar suporte ou sustentar a vida.

**ISENÇÃO DE RESPONSABILIDADE DE APLICAÇÕES DE SEGURANÇA CRÍTICA, MILITAR E AUTOMOTIVA:** Os produtos da Atmel não foram projetados e não serão usados em conexão com quaisquer aplicações em que se espera que a falha de tais produtos resulte em ferimentos pessoais significativos ou morte ("Segurança Crítica Applications") sem o consentimento específico por escrito de um funcionário da Atmel. Aplicações críticas de segurança incluem, sem limitação, dispositivos e sistemas de suporte à vida, equipamentos ou sistemas para a operação de instalações nucleares e sistemas de armas.

Os produtos da Atmel não são projetados nem destinados ao uso em aplicações ou ambientes militares ou aeroespaciais, a menos que especificamente designados pela Atmel como de nível militar. Os produtos da Atmel não são projetados nem destinados ao uso em aplicações automotivas, a menos que especificamente designados pela Atmel como de nível automotivo.