

FACULDADE DE INFORMÁTICA E ADMINISTRAÇÃO
PAULISTA - FIAP
CURSO DE TECNOLOGIA EM DEFESA CIBERNÉTICA

RELATORIO DO VIDEO
TREZOR WALLET POR JOE GRANT

SÃO PAULO – SP

MAIO/2024

GABRIELA BEATRIZ NASCIMENTO SANTOS

RELATORIO DO VIDEO
TREZOR WALLET POR JOE GRANT

ALEXANDRA PERCARIO

SÃO PAULO – SP

MAIO/2024

Criptomoedas uma breve explicação

As criptomoedas são um tipo de moeda digital descentralizada, que não depende de bancos ou governos para sua confiabilidade. Em vez disso, confiam na segurança da criptografia, especialmente na criptografia de chave pública. Cada usuário possui uma chave privada para assinar transações e uma chave pública para verificar transações. Isso garante a segurança e autenticidade das transações sem depender de instituições centralizadas.

Blockchain e Carteiras de Criptomoedas

O blockchain é uma tecnologia essencial para as criptomoedas, funcionando como um livro gigante digital descentralizado. Cada transação é pública e registrada de forma permanente, tornando impossível alterar o registro.

No entanto, a segurança do blockchain depende das chaves privadas dos usuários. No caso de Dan e Jesse, que perderam suas chaves privadas armazenadas em uma carteira de hardware, enfrentaram dificuldades. Uma carteira de hardware é como um computador dedicado que armazena chaves privadas. Se a senha da carteira for esquecida, o acesso aos ativos pode ser perdido permanentemente.

Um exemplo é a carteira rígida Trezor, que apaga o conteúdo da carteira após várias tentativas incorretas de senha. Portanto, é crucial manter as chaves privadas seguras e a senha da carteira em mente para evitar perdas irreparáveis de criptomoedas.

Segurança de Criptomoedas: Lições a serem aprendidas/ Fatos que aconteceram

O caso de pessoas que esqueceram suas senhas, como o de Mark Frounfelder em 2017, destaca a importância da segurança na gestão de criptomoedas. Mark possuía 7,4 bitcoins, avaliados em 3 mil dólares na época, mas perdeu acesso à sua carteira porque sua semente de recuperação foi jogada fora por um faxineiro. Mesmo lembrando do PIN, ele não conseguiu recuperar suas bitcoins, pois percebeu que havia esquecido. Este incidente, que hoje valeria 32 mil dólares, demonstra os desafios enfrentados nos estágios iniciais da criptomoeda.

Outro exemplo ilustrativo é o de um homem que tenta recuperar 280 milhões em bitcoins que acidentalmente jogou fora. Ele tinha 7.500 bitcoins, com a chave privada armazenada em um disco rígido. Por engano, ele jogou fora o disco errado e só percebeu meses depois, quando o disco já estava no depósito de lixo local.

Essas histórias ressaltam a importância de medidas de segurança robustas. Armazenar chaves privadas de forma segura, evitar anotá-las em pedaços de papel ou em dispositivos sem proteção, e estar atento aos procedimentos de recuperação são fundamentais para proteger os ativos de criptomoedas. O caso do homem com 240 milhões de dólares em bitcoins inacessíveis devido à perda de senha reforça a necessidade de precaução e planejamento adequados na gestão de criptomoedas. Estes casos, embora dramáticos, são apenas uma amostra dos desafios enfrentados pelos investidores de criptomoedas e destacam a importância de adotar práticas de segurança eficazes para evitar perdas significativas de ativos digitais.

Vulnerabilidades em Dispositivos Trezor e Implicações na Segurança

O dispositivo Trezor 1 está enfrentando ataques cibernéticos devido ao desenvolvimento contínuo de explorações visando comprometer sua segurança. Ao se deparar com o dispositivo, acreditou que poderia replicar parte do trabalho já realizado por outros. Após três meses de tentativa e erro, conseguiu provocar um mau funcionamento no chip de silício do dispositivo, comprometendo sua segurança.

O resultado foi surpreendente: Conseguiu derrotar a proteção de segurança das informações da semente de recuperação. Embora não fosse exatamente o que ele estava tentando fazer, o PIN necessário apareceu na tela. Esse incidente levou a refletir sobre suas ações e compartilhar a descoberta com sua esposa, que também se mostrou intrigada. Juntos, passaram cerca de uma hora recriando os passos e revisando todo o processo.

Este caso destaca a importância de estar ciente das vulnerabilidades de segurança em dispositivos de armazenamento de criptomoedas. Apesar dos esforços para proteger as informações, é crucial permanecer vigilante e adotar medidas proativas para fortalecer a segurança dos ativos digitais. A descoberta ressalta a necessidade de uma abordagem holística para proteger os investimentos em criptomoedas contra ameaças cibernéticas em constante evolução.

```

144 data2hex(storage_uid, sizeof(storage_uid), storage_uid_str);
145
146 // copy storage
147 size_t old_storage_size = 0;
148
149 if (version == 1 || version == 2) {
150     old_storage_size = 468;
151 } else
152 if (version == 3 || version == 4 || version == 5) {
153     old_storage_size = 1488;
154 } else
155 if (version == 6 || version == 7) {
156     old_storage_size = 1496;
157 } else
158 if (version == 8) {
159     old_storage_size = 1504;
160 }
161
162 memset(&storage, 0, sizeof(Storage));
163 memcpy(&storage, (void *) (FLASH_STORAGE_START + 4 + sizeof(storage_uid)), old_storage_size);
164
165 if (version <= 5) {
166     // convert PIN failure counter from version 5 format
167     uint32_t pinctr = storage.has_pin_failed_attempts
168         ? storage.pin_failed_attempts : 0;
169     if (pinctr > 31) {
170         pinctr = 31;
171         flash_clear_status_flags();
172         flash_unlock();
173         // erase extra storage sector
174         flash_erase_sector(FLASH_META_SECTOR_LAST, FLASH_CR_PROGRAM_X32);
175         flash_program_word(FLASH_STORAGE_PINAREA, 0xffffffff << pinctr);
176         flash_lock();
177         storage_check_flash_errors();
178         storage.has_pin_failed_attempts = false;
179         storage.pin_failed_attempts = 0;
180     }
181 }

```

Para garantir o sucesso do ataque, é necessário empregar diversos hardwares distintos.



Dentro do microcontrolador do dispositivo Trezor, existe um recurso de segurança que impede a leitura do conteúdo da memória. Nossa tarefa consiste em descobrir um método para contornar essa segurança, permitindo-nos acessar a memória. É importante notar que o mecanismo de segurança é ativado apenas durante a inicialização do dispositivo Trezor. Assim, durante nosso ataque, precisamos repetidamente ligar e desligar o Trezor para tentar derrotar essa verificação.

Para realizar esse ciclo, estamos utilizando um dispositivo denominado Phi Whisperer. Este dispositivo é empregado para ligar e desligar o Trezor. A falha só ocorre quando o chip está ligado, sendo essencial ativar o dispositivo para tentar explorar a falha. Caso não tenhamos sucesso, devemos desligar o dispositivo e ligá-lo novamente assim que a energia for aplicada ao Trezor.



Chip Whisperer

O objetivo é contornar a verificação de segurança no momento preciso para enganar o chip, fazendo-o acreditar que temos acesso a ele, mesmo que isso não deva ocorrer. Para alcançar esse objetivo, empregamos uma ferramenta conhecida como Chip Whisperer.



Estamos implementando um ataque de injeção de falhas ou falha de tensão, que visa induzir o comportamento inadequado do chip de forma favorável a nós. Esse método

aproveita a limitação dos sistemas eletrônicos, que operam dentro de parâmetros específicos garantidos pelo fabricante.

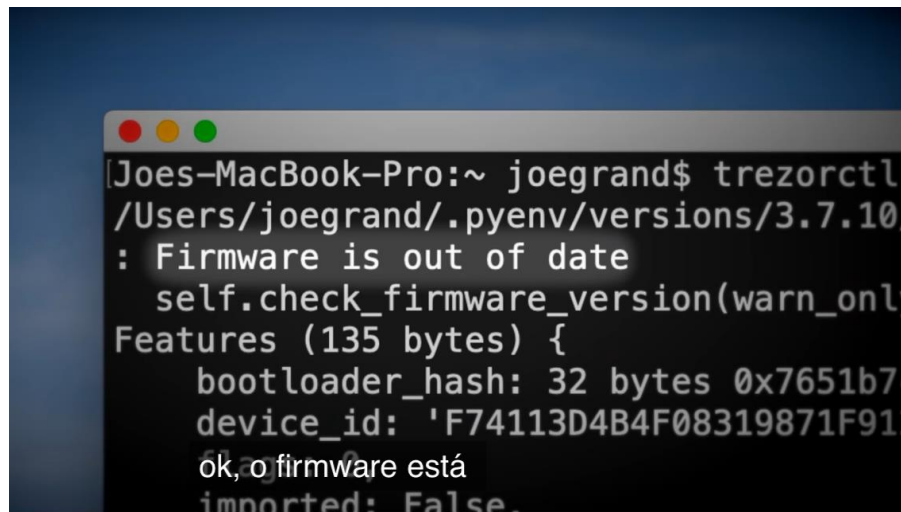
Questionamento

Quando operamos o dispositivo fora dos parâmetros normais, estamos explorando uma falha. Se conseguirmos danificar o chip no momento certo, poderemos superar a segurança e prosseguir com o ataque.

Processo

Para confirmar que derrotamos a segurança, observamos a ativação do modo de depuração pelo chip. Esse modo permite que um engenheiro legítimo leia a memória e realize depurações gerais em um microcontrolador. No caso do Trezor, se conseguirmos superar a segurança, o modo de depuração será ativado, concedendo acesso apenas à área de memória RAM. A recuperação ocorre quando as informações privadas necessárias são copiadas para a RAM. Além disso, é necessário modificar o dispositivo Trezor para conectá-lo ao restante do hardware. Todos os componentes estão interligados por uma placa de circuito personalizada. Se tudo funcionar corretamente, o ataque é bem-sucedido.

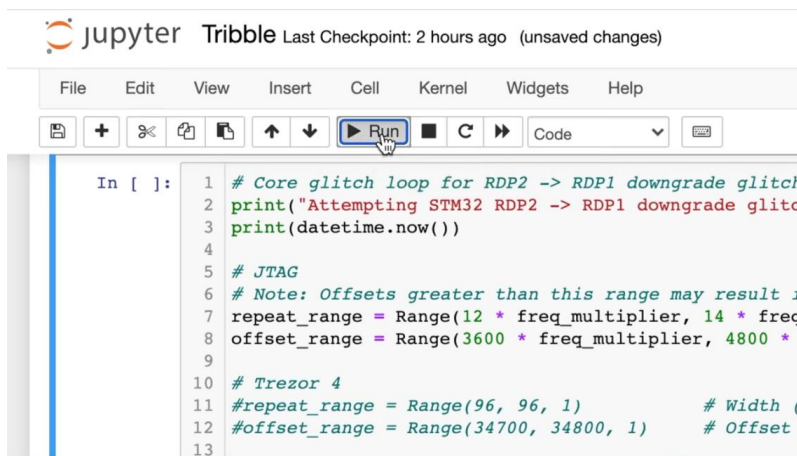
- Conectar o Trezor ao computador e verificar a comunicação.
- Observar o firmware e vê que está desatualizado, versão 1.6.0, o que é vantajoso.

A screenshot of a terminal window on a Mac. The window title is "Joes-MacBook-Pro:~ joegrand\$". The command entered is "trezorctl". The output shows a warning "Firmware is out of date" and a JSON object for "Features (135 bytes)". The JSON includes "bootloader_hash" (32 bytes, 0x7651b7c), "device_id" ('F74113D4B4F08319871F912'), and "ok, o firmware está" (partially visible). The command "imported: False." is also visible at the bottom.

```
Joes-MacBook-Pro:~ joegrand$ trezorctl
/Users/joegrand/.pyenv/versions/3.7.10/
: Firmware is out of date
  self.check_firmware_version(warn_only
Features (135 bytes) {
  bootloader_hash: 32 bytes 0x7651b7c
  device_id: 'F74113D4B4F08319871F912
  ok, o firmware está
imported: False.
```

- Reconhecer o risco inicial de danificar o dispositivo ao abri-lo.
- Realizar cortes leves no dispositivo.

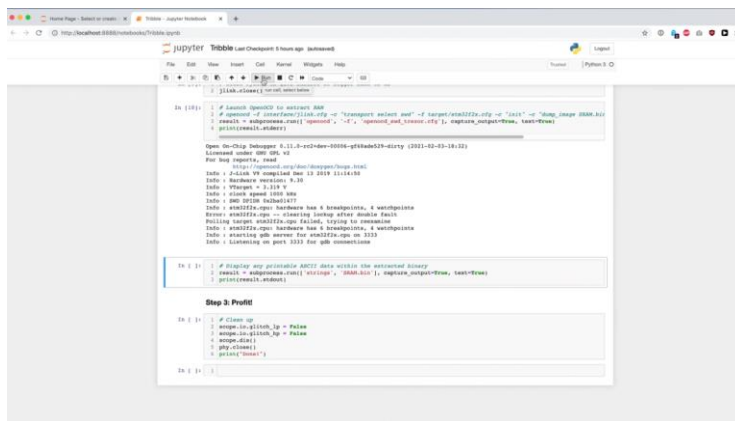
- Identificar o revestimento isolante, uma camada protetora que torna a soldagem das conexões um pouco difícil.
- Aplicar produtos químicos para remover o revestimento e iniciar a soldagem dos fios.
- Utilizar uma caneta removedora de revestimento conformal apenas nas áreas necessárias para garantir uma boa conexão.
- Verificar se o revestimento foi completamente removido usando uma lupa.
- Remover alguns componentes da placa, como capacitores, para tornar o chip mais suscetível a falhas.
- Aquecer cuidadosamente os lados dos componentes para removê-los da placa, evitando danos à placa.
- Adicionar conectores externos para permitir a conexão do dispositivo a outros hardwares.
- Iniciar o processo de ataque de falhas, executando um loop para desligar e ligar o dispositivo e verificar a abertura da interface de depuração.



The screenshot shows a Jupyter Notebook window titled 'Tribble' with a status bar indicating 'Last Checkpoint: 2 hours ago (unsaved changes)'. The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for file operations and execution. The 'Run' button is highlighted. The code cell contains the following Python code:

```
In [ ]: 1 # Core glitch loop for RDP2 -> RDP1 downgrade glitch
2 print("Attempting STM32 RDP2 -> RDP1 downgrade glitch")
3 print(datetime.now())
4
5 # JTAG
6 # Note: Offsets greater than this range may result i
7 repeat_range = Range(12 * freq_multiplier, 14 * freq
8 offset_range = Range(3600 * freq_multiplier, 4800 *
9
10 # Trezor 4
11 #repeat_range = Range(96, 96, 1) # Width (
12 #offset_range = Range(34700, 34800, 1) # Offset
13
```

- Aguardar várias horas, possivelmente até 6 horas, enquanto o processo de ataque passa por cerca de 10.000 tentativas.
- Após 3 horas e 19 minutos, obter acesso à memória.
- Utilizar um programa externo para extrair a memória RAM do dispositivo.



- Copiar a memória RAM e analisar os dados, concluindo assim o processo com o hardware utilizado.

```

Info : stm32f2x.cpu: hardware has 6 breakpoints, 4 watchpoints
Info : starting gdb server for stm32f2x.cpu on 3333
Info : Listening on port 3333 for gdb connections

```

```

In [11]:
1 # Display any printable ASCII data within the extracted binary
2 result = subprocess.run(['strings', 'SRAM.bin'], capture_output=True, text=True)
3 print(result.stdout)

12514
j1 trezor
XXXXXXXXXX
F74113D4B4F08319871F9120
"2:.&

```

Obteve com sucesso a senha do dispositivo.