Decryptors

Documentação – Challenge

FIAP – Defesa Cibernética
São Paulo, 2025

RELATÓRIO DE PROJETO: DETECTOR E BLOQUEADOR DE RANSOMWARES

Grupo Decryptors

Sumário

1. Descrição

O projeto desenvolvido pelo grupo Decryptors consiste em uma aplicação anti-ransomware que monitora em tempo real o sistema de arquivos para identificar possíveis ataques de ransomware antes que eles causem danos significativos.

A solução é estruturada em três camadas principais de defesa:

- **Monitoramento em tempo real:** Utiliza a biblioteca watchdog para detectar modificações e criações de arquivos em diretórios monitorados.
- **Análise de entropia:** Mede o grau de aleatoriedade dos arquivos, usando o cálculo de entropia para identificar arquivos que foram criptografados (entropia alta).
- **Detecção de bursts:** Avalia o número de alterações em uma janela de tempo curta para detectar padrões típicos de ransomware, com rápido aumento no número de arquivos modificados.

Em caso de detecção de comportamento suspeito, a aplicação executa uma **resposta automática** para proteção do sistema, encerrando processos que estejam consumindo recursos elevados de CPU e disco, exceto aqueles pertencentes a uma lista branca de processos legítimos, como aplicativos de compartilhamento de tela.

A interface gráfica foi desenvolvida com Flask (Python), HTML e CSS, permitindo fácil interação para iniciar o monitoramento e visualizar notificações dos eventos detectados.

2. Tecnologias

- **Python:** Linguagem principal para o código de monitoramento, cálculo de entropia, detecção e mitigação.
- **Watchdog:** Biblioteca para monitoramento eficiente de eventos em sistemas de arquivos.
- **Psutil:** Utilizado para gerenciamento e interação com processos do sistema operacional, especialmente para identificar e encerrar processos suspeitos.
- **Flask:** Framework web usado para construção da interface gráfica e disponibilização do monitoramento via browser.
- **HTML/CSS:** Utilizados para criar uma interface responsiva e amigável para o usuário.
- **JavaScript (EventSource):** Implementa comunicação em tempo real para exibir logs e alertas diretamente no navegador.

3. Arquitetura e Componentes

O projeto é composto por três principais módulos:

3.1. Interface Web (app.py)

- Desenvolvida utilizando o framework Flask, oferece uma página inicial e endpoints para inicializar o monitoramento e acessar os logs em tempo real.
- Utiliza multithreading para garantir que o monitoramento e a interface web rodem simultaneamente sem bloquear a aplicação.
- O endpoint /logs implementa um stream server-sent-event, permitindo que os logs sejam acessados e atualizados em tempo real pelo usuário, facilitando o acompanhamento das ações do sistema.

3.2. Núcleo de Detecção e Defesa (entropia.py)

- O monitoramento é realizado a partir do módulo watchdog, atuando sobre um diretório configurável (por padrão, C:/Users/Public). Mudanças em arquivos neste diretório são continuamente avaliadas.
- A detecção combina dois critérios principais:
 • Volume de Mudanças: O sistema mantém um registro temporal dos eventos de criação/modificação. Se o número de eventos ultrapassa um limiar configurável em um curto intervalo (ex: 20 alterações em 10 segundos), um possível ataque de ransomware é detectado.
 • Entropia de Arquivos: Arquivos criados ou modificados passam por uma análise de entropia.

Arquivos com valores elevados indicam criptografia, comportamento típico de ransomwares.
- Em caso de detecção suspeita, o sistema realiza defesa ativa:
• Percorre os processos do sistema e encerra aqueles que consomem alta CPU ou disco, exceto os que estão em uma whitelist associada a aplicações legítimas de compartilhamento de tela (ex: Teams, Zoom).
- Todo o histórico e eventos são registrados em um arquivo de log para auditoria.

3.3. Simulador de Ransomware (ransomware-detector.py)

- Um script separado foi desenvolvido para simular de forma controlada o comportamento de um ransomware: cria arquivos aleatórios carregados de dados binários de alta entropia, em velocidade e volume suficiente para disparar o detector.
- Isso permite testar o sistema de modo seguro, avaliando sua eficiência e capacidade de resposta.

4. Back-end

Arquitetura e Funcionamento

- **Entrada:** O sistema monitora alterações (criação e modificação) em uma pasta alvo definida, capturando eventos gerados.
- **Processamento:**
    - O cálculo da entropia é aplicado a cada arquivo alterado para avaliar se o conteúdo apresenta alta aleatoriedade, típico de arquivos criptografados.
    - O sistema mantém uma lista temporal das alterações e verifica se há um número elevado de modificações (bursts) dentro de uma janela de tempo configurada.
- **Saída:**
    - Logs estruturados são gravados para auditoria, detalhando arquivos afetados e níveis de entropia.
    - Alertas críticos são gerados caso padrões suspeitos sejam detectados.
    - Processo de mitigação automática pode ser ativado para encerrar processos com alto consumo de CPU, excluindo os que estão na lista branca.

5. Considerações

5.1 Benefícios

- Detecção eficaz de ransomware através da análise combinada de entropia e detecção de bursts.
- Integração com processos do sistema para mitigação automática, aumentando a rapidez da resposta.
- Interface web leve e intuitiva que facilita o monitoramento e controle das operações.
- Logs compatíveis para integração futura com sistemas SIEM.

5.2 Limitações

- Potenciais falsos positivos causados por arquivos legítimos com entropia alta, como arquivos comprimidos ou backups.
- Risco de evasão por ransomware que atue lentamente para evitar disparar detecções rápidas.
- Mitigações automáticas requerem atenção para não impactar processos legítimos essenciais ao usuário.

Data de emissão: 02/10/2025