nic.br

SEMANA DE CAPACITAÇÃO

26 a 30 de Setembro de 2022

Edição *Online* 5

inotify: observando os seus arquivos e tomando decisões de segurança em tempo real

João Eriberto Mota Filho

Brasília, DF, 29 de setembro de 2022

Sumário

Sumário

Conceito de inode

- Os inodes fazem parte da área de controle dos filesystems.
- Eles contêm todos os dados sobre um arquivo ou diretório, exceto o seu nome.
- Exemplos de dados existentes nos inodes: tipo de objeto, tamanho do objeto, permissões de acesso, MACtimes, setores de disco ocupados pelo objeto etc.
- Diretórios são arquivos que contêm os nomes de outros arquivos e diretórios, bem como a correlação com os seus números de inode.

Conceito de inode

- O estabelecimento de um nome para um número de inode em diretório se chama hardlink.
- Em resumo, para o sistema, arquivos e diretórios possuem números (o do seu inode principal). Nomes são apenas abstrações para usuários.
- É importante ressaltar, novamente, que cada arquivo ou diretório possui um inode controlador que é numerado.
- Pode ser que haja mais de um inode por arquivo ou diretório. No entanto, será considerado o número do primeiro.

Conceito de inode

- Visualização gráfica, para o arquivo /home/eriberto/teste:

DIRETÓRIO

Diretório: /home/eriberto/

Inode 531222 ---> documentos/
Inode 531689 ---> hotel.jpg
Inode 532021 ---> palestras/
Inode 532455 ---> teste
Inode 534231 ---> xyz.sh

USUÁRIO

Usuário conhece o arquivo teste.
O diretório o liga ao inode 532455 (hardlink).
O inode conhece os dados gerais do arquivo.

INODE

Inode: 532455

Type: file
Size: 2548
Perm.: rwx rw- r--
Owner: eriberto
Group: users

Modify: 09 jul 2015 09:02:23 -0300
Access: 10 jul 2015 05:33:45 -0300
Change: 09 jul 2015 09:02:23 -0300

Sectors: 532 533 534 535 536

Sumário

O que é inotify?

- O inotify (inode notify) é um monitor de eventos de filesystem.
- É parte do Kernel Linux desde a versão 2.6.13 (2005).
- Pode monitorar inodes de arquivos e diretórios.
- Quando o inotify monitora diretórios, ele observa o diretório em si e todos os seus arquivos.
- Resumindo, o inotify monitora mudanças em inodes.
- O Kernel Linux provê uma API para a chamada por parte de programas.

O que é inotify?

- Exemplo real: identificação de mudanças em diretórios por parte de gerenciadores de arquivos.
- Ninguém pensa o que se passa nos bastidores, pois isso já é normal nas nossas vidas.
- Demonstração com shell e Dolphin.

Sumário

Tipos de eventos inotify

- O inotify pode monitorar os seguintes eventos em arquivos e/ou diretórios ($ man 7 inotify):

  IN_ACCESS (+): arq/dir foi acessado.

  IN_ATTRIB (*): metadado foi alterado.

  IN_CLOSE_WRITE (+): arq/dir foi fechado e houve alteração.

  IN_CLOSE_NOWRITE (*): arq/dir foi fechado e não houve alteração.

  IN_CREATE (+): arq/dir criado em um diretório observado.

  IN_DELETE (+): arq/dir deletado em um diretório observado.

  IN_DELETE_ITSELF: arq/dir observado foi deletado.

  (*) Pode ocorrer no diretório ou em objetos dentro dele.
  (+) Ocorre apenas em objetos dentro de diretórios.

Tipos de eventos inotify

- Continuando...

  IN_MODIFY (+): arq/dir teve conteúdo modificado.

  IN_MOVE_SELF: arq/dir observado diretamente foi movido.

  IN_MOVED_FROM (+): arq/dir em um diretório observado foi renomeado. Será mostrado o nome inicial.

  IN_MOVED_TO(+): arq/dir em um diretório observado foi renomeado. Será mostrado o nome final.

  IN_OPEN (*): arq/dir foi aberto.

(*) Pode ocorrer no diretório ou em objetos dentro dele.
(+) Ocorre apenas em objetos dentro de diretórios.

Tipos de eventos inotify

- Há, ainda, as seguintes macros:
  - IN_ALL_EVENTS: equivale a todos os eventos mostrados anteriormente.
  - IN_MOVE: engloba IN_MOVED_FROM e IN_MOVED_TO.
  - IN_CLOSE: engloba IN_CLOSE_WRITE e IN_CLOSE_NOWRITE.
- Há, também, situações especiais. As mais importantes:
  - IN_IGNORED: o evento está ocorrendo em um objeto mas deverá ser ignorado.
  - IN_ISDIR: o evento está ocorrendo em um diretório.

Sumário

Algumas aplicações do inotify

- São as seguintes, algumas possibilidades de uso do inotify:
  - ✓ Backup instantâneo de arquivos, sempre que os mesmos forem alterados.
  - ✓ Remoção de dumps de bancos de dados, imediatamente após o backup para um servidor específico.
  - ✓ A criação de um sistema de DHCP primário e secundário, similar ao que ocorre com o DNS.
  - ✓ Criação de um sistema inteligente de repositório local Debian, para pacotes personalizados, também servindo para outras distribuições.

Algumas aplicações do inotify

- Continuando...
  - ✓ Ok, os repositórios oficiais Debian usam inotify para detectar a chegada de novos pacotes!
  - ✓ Monitoramento de e-mails (mailbox, maildir etc).
  - ✓ Observação de ações de programas sobre o filesystem, podendo atuar como um debug ou verificador de segurança.
  - ✓ Observação de invasões em redes de computadores, obtendo alertas em tempo real, criando-se algo em torno de um elemento verificador de integridade. 😎

Algumas aplicações do inotify

- Continuando...
  - ✓ Gerador de logs, com auxílio do comando logger, para análise em tempo real pelo bloqueador de IPs fail2ban.
  - ✓ Há centenas de outras possibilidades...

Sumário

- Conceito de inode
- O que é inotify?
- Tipos de eventos inotify
- Algumas aplicações do inotify
- **iWatch: um exemplo de uso do inotify**
- Exemplo de configuração do iWatch
- Uso como verificador de integridade
- Outras implementações do inotify
- Conclusão

iWatch: um exemplo de uso do inotify

- O iWatch é um programa em Perl que implementa o inotify, por intermédio da liblinux-inotify2-perl.
- No Debian e derivados, poderá ser instalado facilmente com *# apt install iwatch*.
- Pode ser executado em linha de comando ou como daemon, usando arquivo de configuração.
- Pode atuar recursivamente.
- A linha *$ iwatch -e all_events <objeto>* poderá ser utilizada para observar e aprender sobre eventos.
- Demonstração via linha de comando.

iWatch: um exemplo de uso do inotify

- O iWatch possui dois importantes conjuntos de eventos para uso especial. São eles:
  - ✓ default: é o valor de operação padrão, quando nenhum evento é citado. Corresponde a close_write, create, delete, move, delete_self e move_self.
  - ✓ all_events: corresponde a todos os eventos. Ótimo para debugs e aprendizado sobre eventos.

iWatch: um exemplo de uso do inotify

- Exemplos de linha de comando:
  - Monitorar /tmp, sem recursividade, com eventos default:

    ```
    $ iwatch /tmp
    ```

  - Monitorar /etc, recursivamente, com eventos default. Caso um evento ocorra, enviar um e-mail para admin@foo.bar, com o resultado dos comandos 'w' e 'ps aux' no corpo. O assunto do e-mail será '<nome_do_arquivo> foi alterado':

    ```
    $ iwatch -r -c (w;ps aux) /etc | mail -s '%f  foi alterado' admin@foo.bar
    ```

Sumário

Exemplo de configuração do iWatch

- O iWatch pode ser executado como daemon, usando arquivo de configuração (/etc/iwatch/iwatch.xml).
- A vantagem desse tipo de implementação, além do modo daemon em background, é a possibilidade de monitoramentos múltiplos e complexos.
- A seguir, um exemplo simples de arquivo de configuração.

Exemplo de configuração do iWatch

```
<?xml version="1.0" ?>
<!DOCTYPE config SYSTEM "/etc/iwatch/iwatch.dtd" >

<config>
<guard email="root@example.com" name="iWatch"/>
<watchlist>
  <title>Monitora alteracoes em /etc/shadow</title>
  <contactpoint email="someone@example.com" name="Admin"/>
  <path type="single">/etc/shadow</path>
</watchlist>
</config>
```

Exemplo de configuração do iWatch

- Algumas possibilidades:
  - ✓Vários blocos '<watchlist>', monitorando diversas situações ao mesmo tempo.
  - ✓Uso de scripts externos (shell, Perl etc) como ação após algum evento.
  - ✓Uso de comandos como mail, sendxmpp (mensagens Jabber) e yowsup-cli (mensagens WhatsApp) para o envio de alertas em tempo real.

Sumário

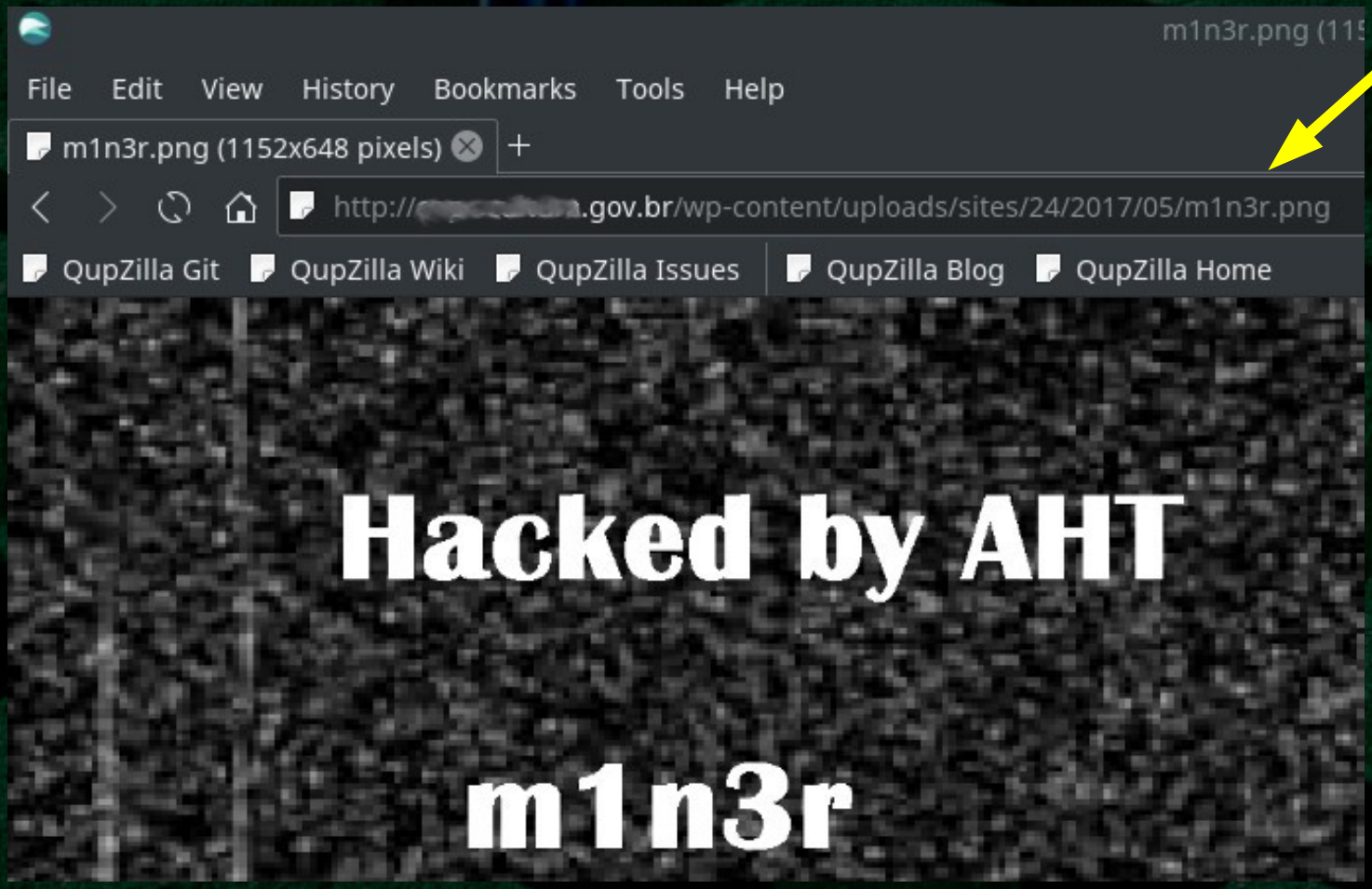
Uso como verificador de integridade

- Sistemas de firewall possuem vários elementos, como filtros de pacotes e estados, proxies, IDS, IPS, etc, etc e verificadores de integridade.
- Os verificadores de integridade observam se houve alterações indesejadas no filesystem.
- A maioria dos verificadores usa checagem de hash via cron e perdem a oportunidade de atuar em tempo real, podendo ser desativados pelos invasores. Ex.: samhain e tripwire.
- Demonstração do iWatch sendo executado como daemon.

Uso como verificador de integridade

Consequência da falta de verificador de integridade...

Uso como verificador de integridade



Injeção

Site original em 26 set. 17.

Mais de 1 mês depois, ninguém sabia do ocorrido...

Uso como verificador de integridade

Mais um...

Uso como verificador de integridade

Situação em 26 set. 2017

(mais de 1 ano depois...)

Uso como verificador de integridade

Mais um...

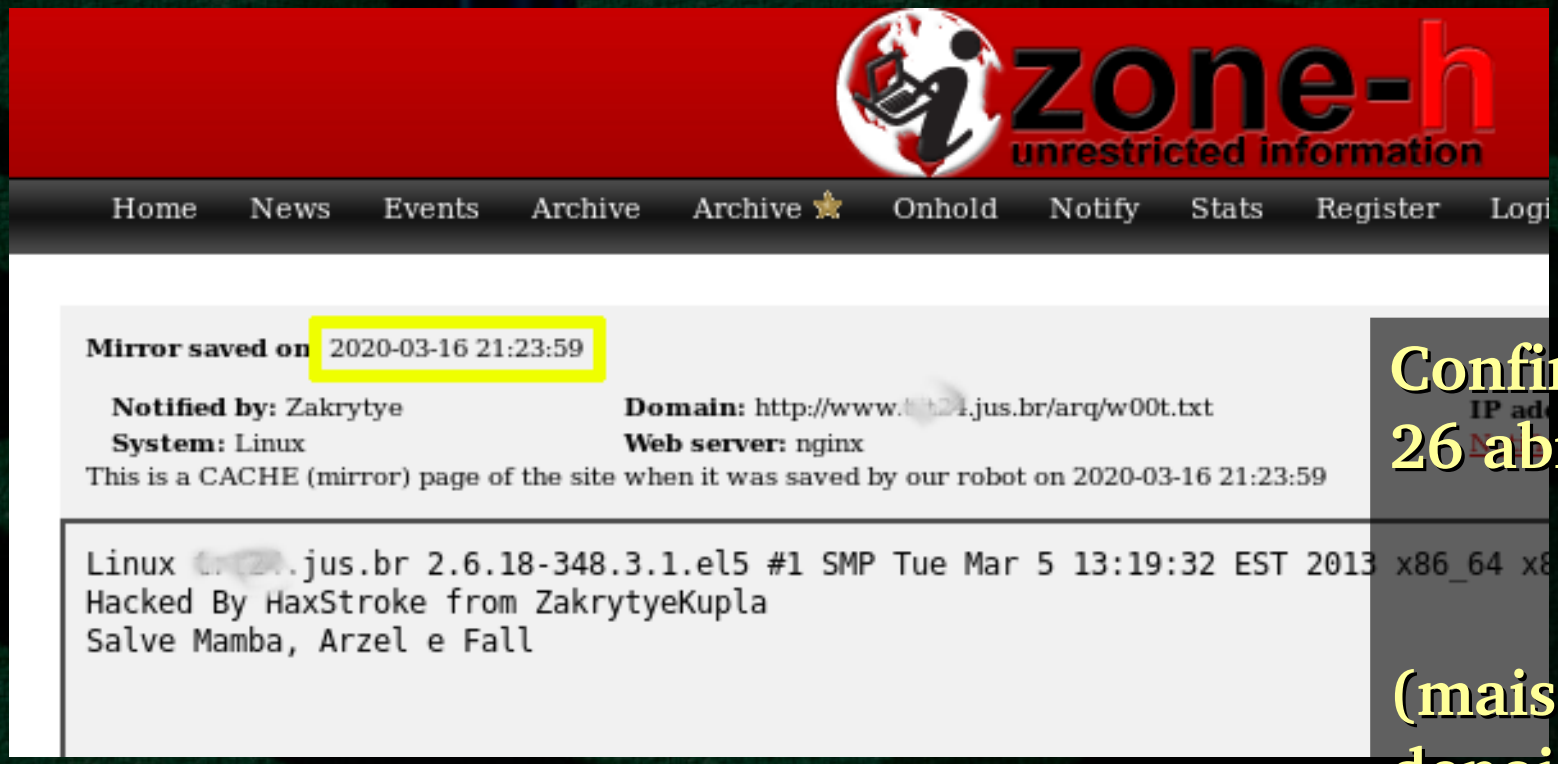
Uso como verificador de integridade

Situação em 26 abr. 2020

(mais de 7 meses depois...)

Uso como verificador de integridade

Outro...



Confirmado em 26 abr. 2020

(mais de 1 mês depois...)

Sumário

Outras implementações do inotify

- Alguns outros programas interessantes baseados em inotify:
  - ✓ clsync: monitor de diretórios leve, especificamente desenvolvido para uso com rsync.
  - ✓ entr (The Event Notify Test Runner): monitor zero-conf extremamente simples. Se houver modificações no(s) arquivo(s), ele executa um comando. Exemplo:

    ```
    $ ls *.pdf | entr echo ok
    ```

  - ✓ gamin: monitor de mudanças em diretórios e filesystems, voltado para a emissão de alertas para ambientes gráficos, como KDE e Gnome.

Outras implementações do inotify

- Alguns outros programas interessantes baseados em inotify:
  - ✓ tail -f (desde 2009).
  - ✓ inoticoming: observa diretórios e realiza ações, caso algum arquivo novo seja criado. É utilizado pelo Debian para processar os pacotes que chegam. Pode observar a chegada de arquivos específicos, como os .change do Debian.
  - ✓ incron: similar a um cron, usando tabelas, baseado em eventos de filesystem em vez de horários.

Outras implementações do inotify

- Alguns outros programas interessantes baseados em inotify:
  - ✓ inotify-tools: conjunto de comandos para a interação com o inotify, permitindo o uso direto por scripts em shell e outras atividades.
  - ✓ inotify-hookable: mais um monitor, similar ao iWatch, prometendo ser melhor do que o inotify-tools.
  - ✓ fswatch: mais um de novo...
  - ✓ fail2ban: observa anomalias em logs e bloqueia endereços IP, com base em falhas de login, tentativas de ataques por força bruta etc.

Outras implementações do inotify

- Alguns outros programas interessantes baseados em inotify:
  - ✓ lsyncd: daemon para sincronizar diretórios locais com auxílio do rsync.
  - ✓ Libraries: há diversas libraries que recebem informações do inotify para uso em aplicações. Há libraries para C, Python, Perl, Haskell, Lua, Ruby, Go etc.
  - ✓ Há uma lista de programas feita em 2019, disponível em https://anarc.at/blog/2019-11-20-file-monitoring-tools/

Sumário

Conclusão

- O inotify provê recursos valiosos para quem precisa monitorar atividades em filesystems.
- O inotify é essencial para quem precisa de monitores de integridade em tempo real em sistemas de firewall.
- Backups inteligentes poderão ser criados facilmente com o auxílio do inotify.

Esta palestra está disponível em:

http://www.eriberto.pro.br

Siga-me em http://twitter.com/eribertomota