

nic.br egi.br

cert.br

Workshop MISP
11 de setembro de 2020
Evento *Online*

MISP Descomplicado: Instalação, Configuração e Operação Básica

Marcus Lahr

Analista de Projetos de Segurança

marcus@cert.br

cert.br **nic.br** **egi.br**

Tratamento de Incidentes

- ▶ Articulação
- ▶ Análise Técnica
- ▶ Apoio à recuperação

Treinamento e Conscientização

- ▶ Cursos
- ▶ Palestras
- ▶ Boas Práticas
- ▶ Reuniões

Análise de Tendências

- ▶ *Honeypots* Distribuídos
- ▶ SpamPots
- ▶ Processamento de *threat feeds*

Filiações e Parcerias:



SEI
Partner
Network



Criação:

Agosto/1996: o relatório “Rumo à Criação de uma Coordenadoria de Segurança de Redes na Internet Brasil” é publicado pelo CGI.br¹

Junho/1997: o CGI.br cria o CERT.br (à época chamado NBSO – *NIC BR Security Office*), com base nas recomendações do relatório²

¹<https://www.nic.br/grupo/historico-gts.htm>

²<https://www.nic.br/pagina/gts/157>

Missão

Aumentar os níveis de segurança e de capacidade de tratamento de incidentes das redes conectadas à Internet no Brasil.

Público Alvo (*Constituency*)

Redes que utilizam recursos alocados pelo NIC.br (endereços IP ou ASNs alocados ao Brasil e domínios sob o ccTLD .br).

Foco das Atividades

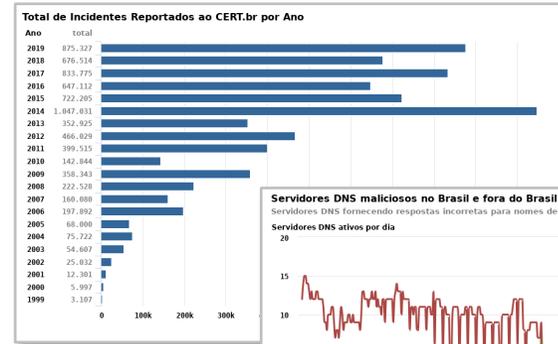
- Ponto de contato nacional
- Trabalho colaborativo com outras entidades
- Auxiliar na análise técnica e compreensão de ataques e ameaças
- Aumentar a detecção, correlação de eventos e determinação de tendências
- Transferir o conhecimento através de cursos, boas práticas e conscientização

Tratamento de Incidentes: Fontes dos Dados, Métricas e Compartilhamento

Notificações voluntárias de incidentes enviadas para:

cert@cert.br

- 2019: 4.086.406 de e-mails tratados, relativos a 875.327 incidentes notificados ao CERT.br



Compartilhamento via MISP

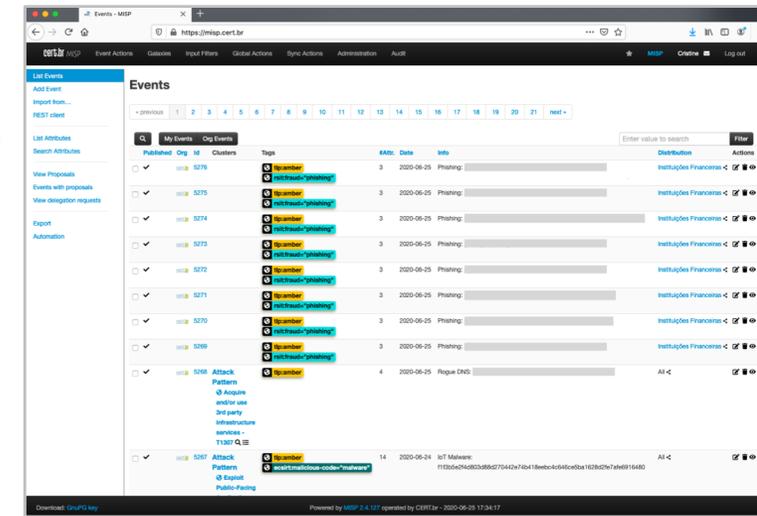
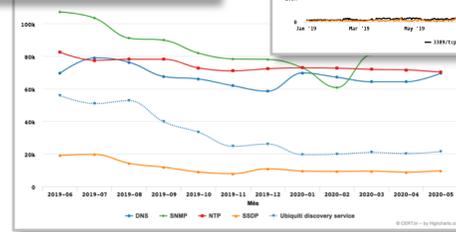
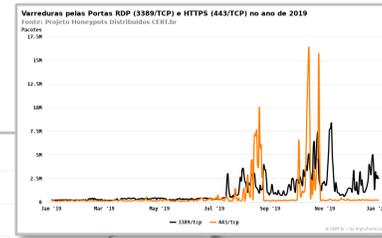
- Indicadores selecionados são compartilhados com parceiros
- Servidores DNS maliciosos
- Phishing
- Binários e Comando e Controle de botnets IoT
- Amplificadores usados em ataques DDoS

Threat feeds

- Honeypots Distribuídos do CERT.br
- Team Cymru
- SpamHaus
- ShadowServer
- Shodan
- Operações Anti-Botnet (Microsoft/FBI)



Notificações para os ASNs e estatísticas públicas



<https://cert.br/stats>

<https://cert.br/misp/>

Agenda

- *Hardening* do SO
- Instalação, configuração e *hardening* dos serviços Web, banco de dados e PHP
- Instalação, configuração e *hardening* do MISP
- Uso do MISP via interface Web
- Uso do MISP de maneira automatizada

Importante: Todos os comandos apresentados nos *slides* estão disponíveis em:

<https://cert.br/misp/tutorial-ubuntu/>



Hardening e **Primeiras Configurações do Sistema Operacional**

cert.br nic.br egi.br

Objetivos

Todo servidor recém instalado é imediatamente vítima de varreduras e ataques.

É essencial, antes de qualquer atualização ou instalação de pacotes, fortalecer a configuração do servidor através de:

- Regras de *firewall* IPv4 e IPv6
- Restrição de acesso ao SSH
 - Somente com chave criptográfica
 - Somente de redes autorizadas

Também faremos configurações básicas de

- *timezone* e sincronia de relógio
- DNS
- Atualizações do sistema operacional

Firewall

As regras sugeridas neste tutorial funcionam da seguinte forma:

- Permitem todo o tráfego *outbound* deste equipamento, permitindo atualização do SO e dos pacotes.
- Obedecem as recomendações da RFC 4890 para filtragem de pacotes ICMPv6.
- Permitem o acesso remoto via SSH apenas por IPs ou redes confiáveis.
- Permitem o acesso ao MISP apenas por IPs ou redes confiáveis.
- Bloqueiam a entrada de pacotes mal formados.
- Permitem testes de TLS via Qualys SSL Labs (opcional).
- Permitem conexões do Let's Encrypt para geração/renovação de certificados (opcional).

Firewall – Adicionando as regras para IPv4 (1/2)

Crie o arquivo `/root/rules.v4` e adicione o seguinte conteúdo:

```
*filter
:INPUT DROP [0:0]
:FORWARD DROP [0:0]
:OUTPUT DROP [0:0]
:IN-NEW - [0:0]
-A INPUT -i lo -j ACCEPT
-A INPUT -p tcp -m tcp ! --tcp-flags FIN,SYN,RST,ACK SYN -m state --state NEW -j DROP
-A INPUT -p tcp -m tcp --tcp-flags FIN,SYN,RST,PSH,ACK,URG NONE -j DROP
-A INPUT -p tcp -m tcp --tcp-flags FIN,SYN FIN,SYN -j DROP
-A INPUT -p tcp -m tcp --tcp-flags SYN,RST SYN,RST -j DROP
-A INPUT -p tcp -m tcp --tcp-flags FIN,RST FIN,RST -j DROP
-A INPUT -p tcp -m tcp --tcp-flags FIN,ACK FIN -j DROP
-A INPUT -p tcp -m tcp --tcp-flags ACK,URG URG -j DROP
-A INPUT -m state --state INVALID -j DROP
-A INPUT -d 224.0.0.0/32 -j REJECT --reject-with icmp-port-unreachable
-A INPUT -p tcp -m state --state ESTABLISHED -j ACCEPT
-A INPUT -p udp -m state --state ESTABLISHED -j ACCEPT
-A INPUT -p icmp -m state --state ESTABLISHED -j ACCEPT
-A INPUT -p icmp -m state --state NEW,ESTABLISHED -m icmp --icmp-type 8 -j ACCEPT
-A INPUT -m state --state NEW -j IN-NEW
-A INPUT -j LOG --log-prefix "IPT_INPUT: " --log-level 6
-A INPUT -j DROP
-A FORWARD -j LOG --log-prefix "IPT_FORWARD: " --log-level 6
-A FORWARD -j DROP
-A OUTPUT -o lo -j ACCEPT
-A OUTPUT -p tcp -m state --state NEW,ESTABLISHED -j ACCEPT
-A OUTPUT -p udp -m state --state NEW,ESTABLISHED -j ACCEPT
-A OUTPUT -p icmp -m state --state NEW,ESTABLISHED -j ACCEPT
-A OUTPUT -j LOG --log-prefix "IPT_OUTPUT: " --log-level 6
-A OUTPUT -j DROP
```

[CONTINUA NO PRÓXIMO SLIDE]

Firewall – Adicionando as regras para IPv4 (2/2)

[CONTINUAÇÃO DO SLIDE ANTERIOR]

```
#liberar acesso SSH para a organização
```

```
-A IN-NEW -s <SEU-IPv4-OU-REDE> -p tcp -m tcp --dport 22 --tcp-flags FIN,SYN,RST,ACK SYN -j ACCEPT
```

```
#liberar acesso ao MISP para a organização
```

```
-A IN-NEW -s <SEU-IPv4-OU-REDE> -p tcp -m tcp --tcp-flags FIN,SYN,RST,ACK SYN -m multiport --dports 80,443 -j ACCEPT
```

```
#liberar acesso ao MISP para parceiros
```

```
-A IN-NEW -s <IPv4-MISP-PARCEIRO> -p tcp -m tcp --tcp-flags FIN,SYN,RST,ACK SYN -m multiport --dports 80,443 -j ACCEPT
```

```
#IPs let's encrypt (opcional)
```

```
-A IN-NEW -s 66.133.109.36/32 -p tcp -m tcp --tcp-flags FIN,SYN,RST,ACK SYN --dport 80 -j ACCEPT
```

```
#IPs sslabs (opcional)
```

```
-A IN-NEW -s 64.41.200.0/24 -p tcp -m tcp --tcp-flags FIN,SYN,RST,ACK SYN --dport 443 -j ACCEPT
```

```
COMMIT
```

Firewall – Adicionando as regras para IPv6 (1/2)

Crie o arquivo `/root/rules.v6` e adicione o seguinte conteúdo:

```
*filter
:INPUT DROP [0:0]
:FORWARD DROP [0:0]
:OUTPUT DROP [0:0]
-A INPUT -i lo -j ACCEPT
-A INPUT -m rt --rt-type 0 -j DROP
-A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A INPUT -m conntrack --ctstate INVALID -j DROP
-A INPUT -p ipv6-icmp -m icmp6 --icmpv6-type 1 -j ACCEPT
-A INPUT -p ipv6-icmp -m icmp6 --icmpv6-type 2 -j ACCEPT
-A INPUT -p ipv6-icmp -m icmp6 --icmpv6-type 3 -j ACCEPT
-A INPUT -p ipv6-icmp -m icmp6 --icmpv6-type 4 -j ACCEPT
-A INPUT -p ipv6-icmp -m icmp6 --icmpv6-type 128 -j ACCEPT
-A INPUT -p ipv6-icmp -m icmp6 --icmpv6-type 129 -j ACCEPT
-A INPUT -p ipv6-icmp -m icmp6 --icmpv6-type 133 -j ACCEPT
-A INPUT -p ipv6-icmp -m icmp6 --icmpv6-type 134 -j ACCEPT
-A INPUT -p ipv6-icmp -m icmp6 --icmpv6-type 135 -j ACCEPT
-A INPUT -p ipv6-icmp -m icmp6 --icmpv6-type 136 -j ACCEPT
-A INPUT -p ipv6-icmp -m icmp6 --icmpv6-type 141 -j ACCEPT
-A INPUT -p ipv6-icmp -m icmp6 --icmpv6-type 142 -j ACCEPT
-A INPUT -p ipv6-icmp -m icmp6 --icmpv6-type 148 -j ACCEPT
-A INPUT -p ipv6-icmp -m icmp6 --icmpv6-type 149 -j ACCEPT
```

[CONTINUA NO PRÓXIMO SLIDE]

Firewall – Adicionando as regras para IPv6 (2/2)

[CONTINUAÇÃO DO SLIDE ANTERIOR]

```
#liberar acesso SSH para a organização
```

```
-A INPUT -s <SEU-IPv6-OU-REDE> -p tcp -m tcp --dport 22 -j ACCEPT
```

```
#liberar acesso ao MISP para a organização
```

```
-A INPUT -s <SEU-IPv6-OU-REDE> -p tcp -m tcp -m multiport --dports 80,443 -j ACCEPT
```

```
#liberar acesso ao MISP para parceiros
```

```
-A INPUT -s <IPv6-MISP-PARCEIRO> -p tcp -m tcp -m multiport --dports 80,443 -j ACCEPT
```

```
#IPs let's encrypt (opcional)
```

```
-A INPUT -s 2600:3000::/29 -p tcp -m tcp --dport 80 --tcp-flags FIN,SYN,RST,ACK SYN -j ACCEPT
```

```
-A INPUT -s 2600:1f00::/24 -p tcp -m tcp --dport 80 --tcp-flags FIN,SYN,RST,ACK SYN -j ACCEPT
```

```
-A INPUT -s 2a05:d000::/25 -p tcp -m tcp --dport 80 --tcp-flags FIN,SYN,RST,ACK SYN -j ACCEPT
```

```
#IPs sslabs (opcional)
```

```
-A INPUT -s 2600:C02:1020:4202::/64 -p tcp -m tcp --dport 443 -j ACCEPT
```

```
-A INPUT -j LOG --log-prefix "IPT_INPUT6: " --log-level 6
```

```
-A INPUT -j REJECT --reject-with icmp6-port-unreachable
```

```
-A FORWARD -j REJECT --reject-with icmp6-port-unreachable
```

```
-A OUTPUT -j ACCEPT
```

```
COMMIT
```

Firewall – Carregando as regras

Carregue as regras com os comandos:

```
# iptables-restore /root/rules.v4  
# ip6tables-restore /root/rules.v6
```

Verifique se as regras foram carregadas com os comandos:

```
# iptables -nL  
# ip6tables -nL
```

Firewall – Tornando as regras permanentes

Para garantir que após um *reboot* as regras atuais sejam carregadas novamente, é necessário instalar o pacote `iptables-persistent`:

```
# apt-get update
# apt-get install iptables-persistent -y
```

Configuring iptables-persistent

Current iptables rules can be saved to the configuration file `/etc/iptables/rules.v4`. These rules will then be loaded automatically during system startup.

Rules are only saved automatically during package installation. See the manual page of `iptables-save(8)` for instructions on keeping the rules file up-to-date.

Save current IPv4 rules?

<Yes>

<No>

Configuring iptables-persistent

Current iptables rules can be saved to the configuration file `/etc/iptables/rules.v6`. These rules will then be loaded automatically during system startup.

Rules are only saved automatically during package installation. See the manual page of `ip6tables-save(8)` for instructions on keeping the rules file up-to-date.

Save current IPv6 rules?

<Yes>

<No>

Responda **<Yes>** para as duas perguntas

Firewall – Verificando as regras

Verifique se as regras foram criadas no diretório `/etc/iptables/` com o comando:

```
# ls -la /etc/iptables/  
total 16  
drwxr-xr-x  2 root root 4096 Aug 28 17:55 .  
drwxr-xr-x 91 root root 4096 Aug 28 17:55 ..  
-rw-r--r--  1 root root 1980 Aug 28 17:55 rules.v4  
-rw-r--r--  1 root root 1907 Aug 28 17:55 rules.v6
```

SSH

- O acesso ao servidor via SSH será configurado para utilizar um segundo fator de autenticação
- Para este segundo fator serão utilizados o par de chaves pública/privada
- O *login* por senha será desabilitado

SSH – Gerando chaves em máquinas *UNIX-Like*

gere o par de chaves com o comando `ssh-keygen`:

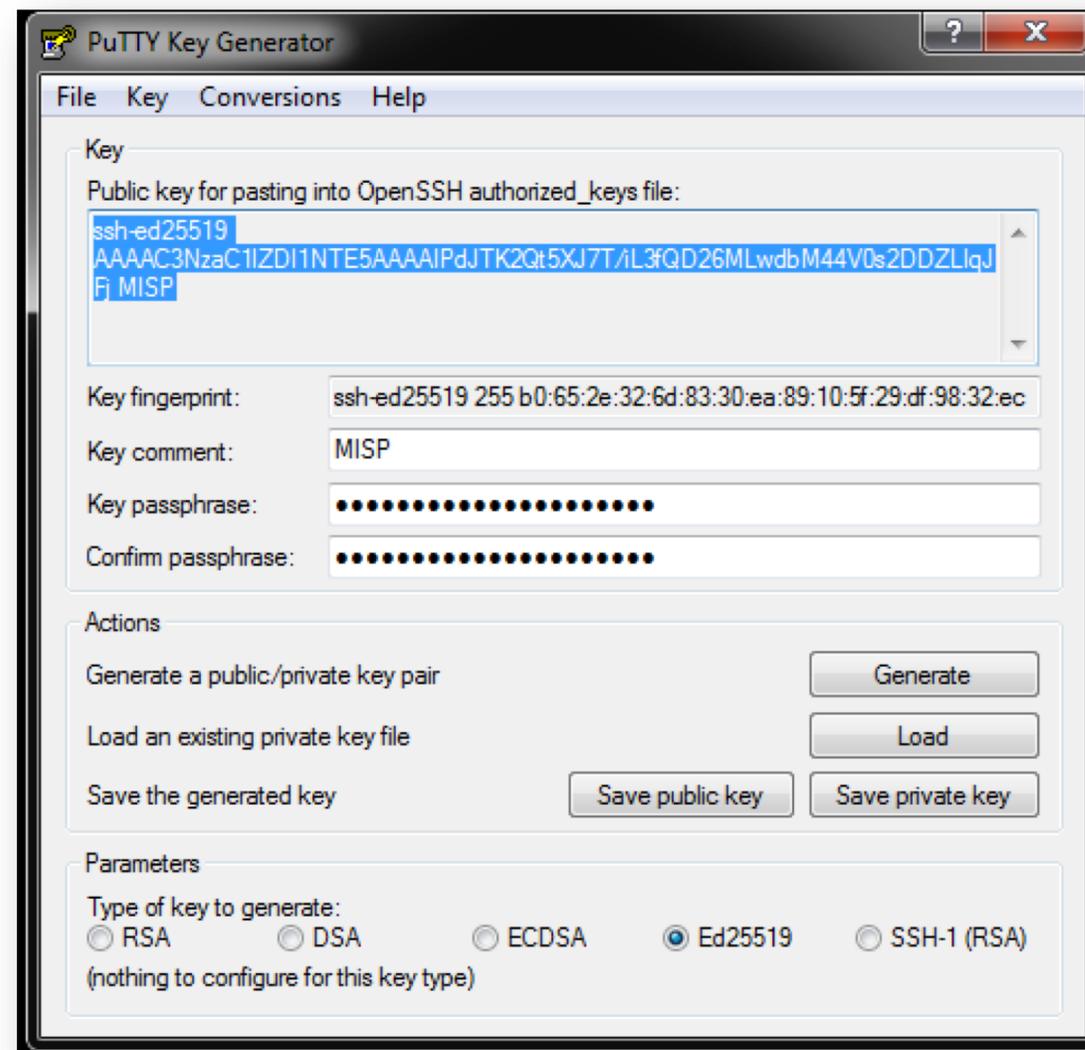
```
# ssh-keygen -t ed25519 -q -f /path/da/chave/misp_ed25519 -C 'MISP'
```

Inclua o conteúdo do arquivo da chave pública recém gerada (`misp_ed25519.pub`) no arquivo `/root/.ssh/authorized_keys` do seu servidor MISP

SSH – Gerando chaves em máquinas Windows

Gere o par de chaves com o aplicativo PuTTY Key Generator (`puttygen.exe`):

- Escolha o tipo `Ed25519`
- Utilize senhas fortes
- Copie o conteúdo da caixa “Public key for pasting into OpenSSH authorized_keys file” para o arquivo `/root/.ssh/authorized_keys` do seu servidor MISP



SSH – Configurando o servidor

- Configure o *daemon* do SSH editando o arquivo `/etc/ssh/sshd_config` do servidor MISP, alterando se necessário o conteúdo das seguintes linhas:

```
PermitRootLogin prohibit-password  
PubkeyAuthentication yes  
PasswordAuthentication no
```

- Reinicie o serviço do SSH com o comando:

```
# service sshd restart
```

Configurando o *timezone*

Configure o *timezone* para UTC com o comando:

```
# timedatectl set-timezone UTC
```

Verifique se o *timezone* foi configurado para UTC com o comando:

```
# timedatectl
```

```
Local time: Fri 2020-08-14 19:41:04 UTC
          Universal time: Fri 2020-08-14 19:41:04 UTC
          RTC time: Fri 2020-08-14 19:41:05
          Time zone: UTC (UTC, +0000)  <-- Indica que o servidor está em UTC
System clock synchronized: yes
systemd-timesyncd.service active: yes
          RTC in local TZ: no
```

Configurando o serviço NTP

Configure o serviço NTP editando o arquivo `/etc/systemd/timesyncd.conf` descomentando e alterando as seguintes linhas:

```
NTP=a.ntp.br
```

```
FallbackNTP=b.ntp.br
```

Reinicie o serviço `timesyncd` com o comando:

```
# service systemd-timesyncd restart
```

Testando o serviço NTP

Verifique se o serviço está rodando com o comando:

```
# service systemd-timesyncd status
```

```
systemd-timesyncd.service - Network Time Synchronization
Loaded: loaded (/lib/systemd/system/systemd-timesyncd.service; enabled; vendor preset: enabled)
Active: active (running) since Tue 2020-08-18 14:50:37 UTC; 6h ago
Docs: man:systemd-timesyncd.service(8)
Main PID: 31091 (systemd-timesyn)
Status: "Synchronized to time server 200.160.0.8:123 (a.ntp.br)."  
Tasks: 2 (limit: 1107)
CGroup: /system.slice/systemd-timesyncd.service
└─31091 /lib/systemd/systemd-timesyncd
```

```
Aug 18 14:50:37 servername systemd[1]: Starting Network Time Synchronization...
```

```
Aug 18 14:50:37 servername systemd[1]: Started Network Time Synchronization.
```

```
Aug 18 14:50:37 servername systemd-timesyncd[31091]: Synchronized to time server [2001:12ff::8]:123 (a.ntp.br).
```

```
Aug 18 15:23:43 servername systemd-timesyncd[31091]: Synchronized to time server 200.160.0.8:123 (a.ntp.br).
```

Servidor DNS Recursivo

Um servidor recursivo local é útil para não depender de um serviço de terceiros, que pode estar indisponível e/ou comprometido.

Adicionalmente diminui a latência nas consultas e é capaz de validar DNSSEC.

Será utilizado o servidor unbound para este fim.

Unbound – Instalação e configuração

Instale o unbound com o comando:

```
# apt-get install unbound
```

Desabilite o *resolver* do sistema com os comandos:

```
# systemctl disable systemd-resolved
```

```
# systemctl stop systemd-resolved
```

```
# rm /etc/resolv.conf
```

Crie novamente o arquivo `/etc/resolv.conf` e adicione o seguinte conteúdo:

```
nameserver ::1
```

```
nameserver 127.0.0.1
```

Reinicie o serviço do unbound com o comando:

```
# service unbound restart
```

Unbound – Resolução de nome validando DNSSEC

Teste a resolução de nomes com o comando:

```
# dig www.dnssec-failed.org @127.0.0.1

; <<>> DiG 9.11.3-1ubuntu1.13-Ubuntu <<>> www.dnssec-failed.org
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 6943
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.dnssec-failed.org. IN A

;; Query time: 502 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Thu Sep 10 02:08:51 UTC 2020
;; MSG SIZE rcvd: 50
```

Unbound – Resolução de nome não validando DNSSEC

Se o unbound não estiver validando DNSSEC, o resultado será semelhante a este:

```
# dig www.dnssec-failed.org @127.0.0.1

; <<>> DiG 9.11.3-1ubuntu1.13-Ubuntu <<>> www.dnssec-failed.org
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 31136
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.dnssec-failed.org. IN A

;; ANSWER SECTION:
www.dnssec-failed.org. 7200 IN A 68.87.109.242
www.dnssec-failed.org. 7200 IN A 69.252.193.191

;; Query time: 180 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Thu Sep 10 02:09:50 UTC 2020
;; MSG SIZE rcvd: 82
```

Mantendo o sistema atualizado

- É recomendado manter o sistema operacional e suas aplicações sempre atualizadas.
- O pacote `cron-apt` ajuda nesta tarefa, pois envia um *e-mail* para o administrador do sistema sempre que uma atualização estiver disponível.
- Para o envio dos *e-mails* é necessário que o servidor tenha um MTA instalado.

Atualização do Ubuntu

Atualize o Ubuntu com os comandos:

```
# apt-get update
```

```
# apt-get dist-upgrade -y
```

Reinicie o servidor se necessário

Instalando o postfix

- Importante para o envio de *e-mails* para o administrador do sistema
- Instale com o comando:

```
# apt-get install postfix -qy
```
- Configure ele para ouvir apenas em *localhost*
- Adicione o *e-mail* do administrador em `/etc/aliases`
 - Crie uma entrada

```
root: <email do administrador>
```
- Rode o comando `newaliases` para recarregar este arquivo
- Mais informações em <https://cert.br/misp/tutorial-ubuntu/>

Instalando o cron-apt

Instale o pacote cron-apt com o comando:

```
# apt-get install cron-apt -qy
```

Edite o arquivo /etc/cron-apt/config e insira o seguinte conteúdo:

```
MAILTO="root"  
MAILON="upgrade"  
OPTIONS="-o Acquire::http::Dl-Limit=125"  
### EOF
```

Reinicie o serviço do cron com o comando:

```
# service cron restart
```

Instalação do MariaDB, Apache, PHP e outras dependências do MISP

cert.br nic.br egi.br

Objetivos

Instalar e configurar os serviços necessários para o funcionamento do MISP:

- Servidor de banco de dados: MariaDB
- Servidor Web: Apache
- PHP
- Outros pacotes essenciais para o funcionamento do MISP

Instalando o MariaDB

Instale o MariaDB com o seguinte comando:

```
# apt-get install mariadb-client mariadb-server -qy
```

MariaDB - *Hardening*

Utilize o `mysql_secure_installation` para fazer o *hardening* do banco de dados.

O `mysql_secure_installation` é um *shell script* instalado junto com o MariaDB. Ao ser executado ele permite realizar as seguintes operações:

- Alterar a senha do usuário root
- Garantir que o usuário root acesse o banco de dados via *localhost* apenas
- Remover as contas anônimas
- Remover o banco de dados de teste, que por *default* pode ser acessado por qualquer usuário

Execute o *hardening* com o comando:

```
# mysql_secure_installation
```

Sugestão para gerar a senha de root:

```
# openssl rand -base64 15
```

Checando o MariaDB

Confira se o MariaDB está rodando com o comando:

```
# systemctl status mariadb
```

```
● mariadb.service - MariaDB 10.1.44 database server
Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor preset: enabled)
Active: active (running) since Mon 2020-08-10 13:01:46 UTC; 9min ago
Docs: man:mysql(8)
      https://mariadb.com/kb/en/library/systemd/
Main PID: 27147 (mysqld)
Status: "Taking your SQL requests now..."
Tasks: 27 (limit: 1107)
CGroup: /system.slice/mariadb.service
└─27147 /usr/sbin/mysqld

Aug 10 13:01:46 servername /etc/mysql/debian-start[27184]: information_schema
Aug 10 13:01:46 servername /etc/mysql/debian-start[27184]: mysql
Aug 10 13:01:46 servername /etc/mysql/debian-start[27184]: performance_schema
Aug 10 13:01:46 servername /etc/mysql/debian-start[27184]: Phase 6/7: Checking and upgrading tables
Aug 10 13:01:46 servername /etc/mysql/debian-start[27184]: Processing databases
Aug 10 13:01:46 servername /etc/mysql/debian-start[27184]: information_schema
Aug 10 13:01:46 servername /etc/mysql/debian-start[27184]: performance_schema
Aug 10 13:01:46 servername /etc/mysql/debian-start[27184]: Phase 7/7: Running 'FLUSH PRIVILEGES'
Aug 10 13:01:46 servername /etc/mysql/debian-start[27184]: OK
Aug 10 13:01:47 servername /etc/mysql/debian-start[27246]: Triggering myisam-recover for all MyISAM tables and
aria-recover for all Aria tables
```

Instalando o Apache

Instale o apache com o comando:

```
# apt-get install apache2 \
apache2-doc apache2-utils -qy
```

Edite o arquivo

```
/etc/apache2/sites-available/000-
default.conf
```

Descomente a linha `ServerName` e preencha com seu FQDN

Recarregue o apache com o comando:

```
# service apache2 reload
```

```
<VirtualHost *:80>
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
ServerName <FQDN>

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

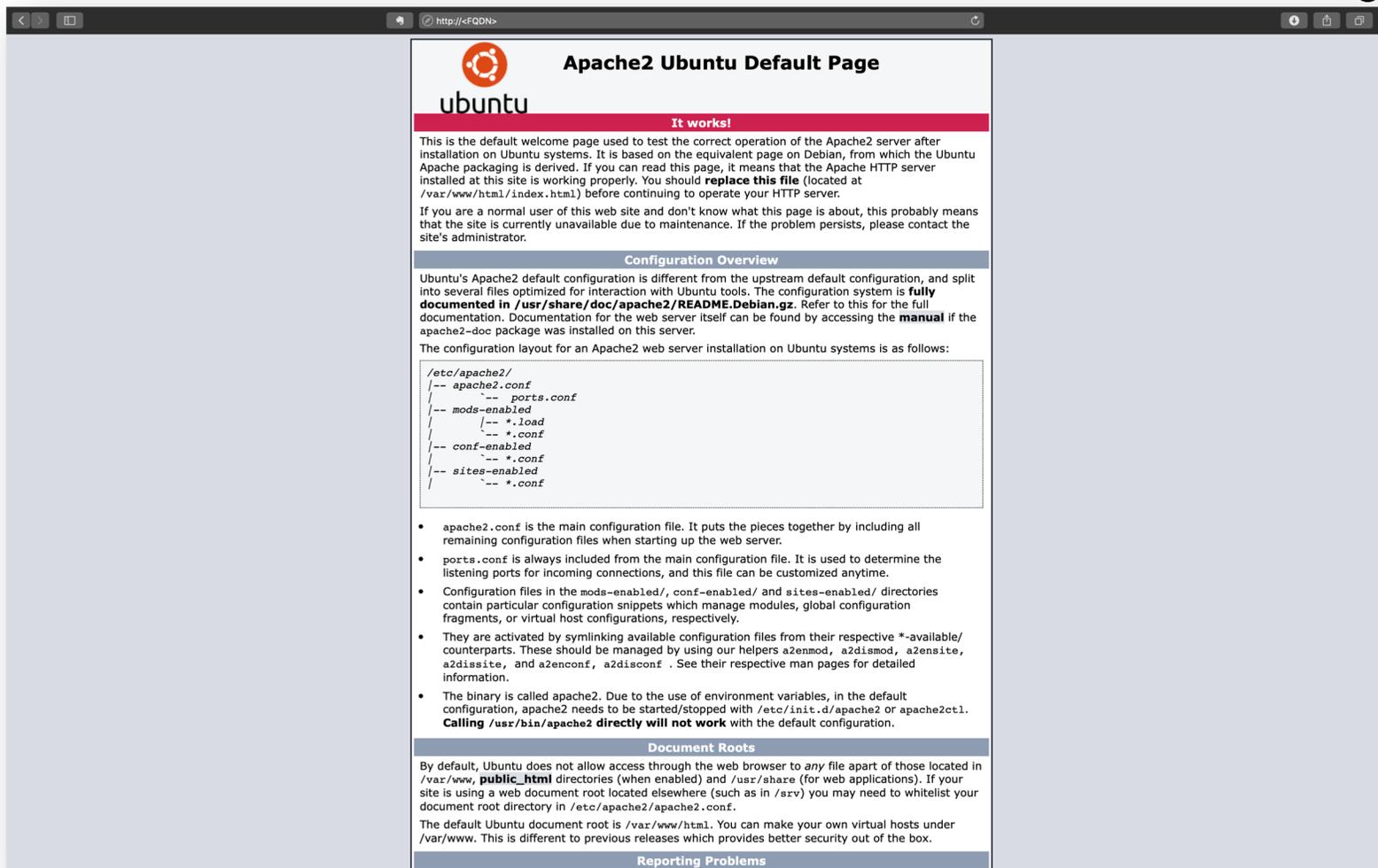
# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
~
~
~
~
~
```

Testando o apache

Verifique se o servidor apache está funcionando acessando seu FQDN através de um navegador

- `http://<FQDN>/`



The screenshot shows a web browser window displaying the Apache2 Ubuntu Default Page. The page title is "Apache2 Ubuntu Default Page" and it features the Ubuntu logo. A red banner with the text "It works!" is prominently displayed. Below this, a paragraph explains that this is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It mentions that the page is based on the equivalent page on Debian and that if the user can read this page, it means the Apache HTTP server is working properly. A note advises replacing the file located at `/var/www/html/index.html` before continuing to operate the HTTP server. A second paragraph states that if the user is a normal user and doesn't know what the page is about, it probably means the site is currently unavailable due to maintenance, and they should contact the site's administrator.

The page includes a "Configuration Overview" section, which explains that Ubuntu's Apache2 default configuration is different from the upstream default configuration and is split into several files optimized for interaction with Ubuntu tools. It states that the configuration system is fully documented in `/usr/share/doc/apache2/README.Debian.gz` and refers to this for the full documentation. It also mentions that documentation for the web server itself can be found by accessing the manual if the `apache2-doc` package was installed on this server. The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/  
|-- apache2.conf  
|   |-- ports.conf  
|-- mods-enabled  
|   |-- *.load  
|   |-- *.conf  
|-- conf-enabled  
|   |-- *.conf  
|-- sites-enabled  
|   |-- *.conf
```

A bulleted list provides details about the configuration files:

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective `*-available/` counterparts. These should be managed by using our helpers `a2enmod`, `a2dismod`, `a2ensite`, `a2dissite`, and `a2enconf`, `a2disconf`. See their respective man pages for detailed information.
- The binary is called `apache2`. Due to the use of environment variables, in the default configuration, `apache2` needs to be started/stopped with `/etc/init.d/apache2` or `apache2ctl`. **Calling `/usr/bin/apache2` directly will not work** with the default configuration.

The "Document Roots" section explains that by default, Ubuntu does not allow access through the web browser to any file apart of those located in `/var/www`, `public_html` directories (when enabled) and `/usr/share` (for web applications). If your site is using a web document root located elsewhere (such as in `/srv`) you may need to whitelist your document root directory in `/etc/apache2/apache2.conf`. The default Ubuntu document root is `/var/www/html`. You can make your own virtual hosts under `/var/www`. This is different to previous releases which provides better security out of the box.

The page concludes with a "Reporting Problems" section.

Let's Encrypt

Instalação do certbot e geração de certificado (1/2)

Se sua organização já possui um certificado digital este passo não é necessário!

Adicione o repositório do certbot com o comando: (necessário apenas no Ubuntu 18.04 LTS)

```
# add-apt-repository ppa:certbot/certbot
```

Instale o pacote do certbot com o comando:

```
# apt-get install certbot python3-certbot-apache -qy
```

Mais informações sobre o certbot: <https://certbot.eff.org/>

Let's Encrypt

Instalação do certbot e geração de certificado (2/2)

Crie um novo certificado com o comando:

```
# certbot certonly --apache
```

- Digite seu endereço de e-mail quando solicitado
- Concorde com os termos de serviço
- Aceite ou não compartilhar seu e-mail com a EFF (opcional)
- Na pergunta: "Which names would you like to activate HTTPS for?" Escolha o número que mostra seu FQDN.



```
Which names would you like to activate HTTPS for?  
-----  
1: <seu FQDN>  
-----  
Select the appropriate numbers separated by commas and/or spaces, or leave input  
blank to select all options shown (Enter 'c' to cancel): 1
```

O certificado será gerado em:

```
/etc/letsencrypt/live/<seu FQDN>/fullchain.pem
```

A chave privada do certificado será gerada em

```
/etc/letsencrypt/live/<seu FQDN>/privkey.pem
```

Let's Encrypt

Renovação de certificado

Os certificados emitidos pela Let's Encrypt tem validade de 90 dias.

A Let's Encrypt recomenda a renovação deste certificado a cada 60 dias.

No Ubuntu o certbot instala um *script* para renovação em `/etc/cron.d/certbot`

Este script roda a cada 12 horas e renova automaticamente qualquer certificado que expire dentro de 30 dias.

A renovação do certificado pode ser testada com o comando:

```
# certbot renew --dry-run
```

```
-----  
** DRY RUN: simulating 'certbot renew' close to cert expiry  
**           (The test certificates below have not been saved.)  
  
Congratulations, all renewals succeeded. The following certs have been renewed:  
/etc/letsencrypt/live/<seu FQDN>/fullchain.pem (success)  
** DRY RUN: simulating 'certbot renew' close to cert expiry  
**           (The test certificates above have not been saved.)  
-----
```

Instalando o PHP

Adicione o repositório do PHP para garantir a instalação do PHP 7.4 (necessário apenas no Ubuntu 18.04 LTS):

```
# add-apt-repository ppa:ondrej/php
```

Instale o PHP e módulos com o comando:

```
# apt-get install libapache2-mod-php7.4 php php-cli php-dev \  
php-json php-xml php-mysql php7.4-opcache php-readline \  
php-mbstring php-redis php-gnupg php-gd -qy
```

Configurando o PHP

Edite o arquivo `/etc/php/7.4/apache2/php.ini` e altere os valores padrão pelos valores sugeridos pelo MISP:

```
upload_max_filesize=50M
post_max_size=50M
max_execution_time=300
memory_limit=2048M
```

Verifique se as diretivas `expose_php` e `display_errors` estão desligadas:

```
expose_php=Off
display_errors=Off
```

Testando o PHP

Teste o PHP criando o arquivo `/var/www/html/phpinfo.php` com o seguinte conteúdo:

```
<?php
    phpinfo()
?>
```

Acesse `http://<FQDN>/phpinfo.php` e verifique se o PHP está funcionando.

Ao final do teste, remova o arquivo `phpinfo.php` com o comando:

```
# rm /var/www/html/phpinfo.php
```

The screenshot shows a web browser window displaying the output of the `phpinfo()` function. The page title is "PHP Version 7.4.9" and it features the PHP logo. The content is organized into several sections:

- System:** Linux misp 4.15.0-115-generic #116-Ubuntu SMP Wed Aug 26 14:04:49 UTC 2020 x86_64
- Build Date:** Aug 7 2020 14:29:36
- Server API:** Apache 2.0 Handler
- Virtual Directory Support:** disabled
- Configuration File (php.ini) Path:** /etc/php/7.4/apache2
- Loaded Configuration File:** /etc/php/7.4/apache2/php.ini
- Scan this dir for additional .ini files:** /etc/php/7.4/apache2/conf.d
- Additional .ini files parsed:** A long list of configuration files including /etc/php/7.4/apache2/conf.d/10-mysqld.ini, /etc/php/7.4/apache2/conf.d/10-opcache.ini, /etc/php/7.4/apache2/conf.d/10-pdo.ini, /etc/php/7.4/apache2/conf.d/15-xsl.ini, /etc/php/7.4/apache2/conf.d/20-calendar.ini, /etc/php/7.4/apache2/conf.d/20-ctype.ini, /etc/php/7.4/apache2/conf.d/20-dom.ini, /etc/php/7.4/apache2/conf.d/20-exif.ini, /etc/php/7.4/apache2/conf.d/20-ftp.ini, /etc/php/7.4/apache2/conf.d/20-geoip.ini, /etc/php/7.4/apache2/conf.d/20-gettext.ini, /etc/php/7.4/apache2/conf.d/20-gnupg.ini, /etc/php/7.4/apache2/conf.d/20-iconv.ini, /etc/php/7.4/apache2/conf.d/20-igmp.ini, /etc/php/7.4/apache2/conf.d/20-json.ini, /etc/php/7.4/apache2/conf.d/20-mbstring.ini, /etc/php/7.4/apache2/conf.d/20-mysql.ini, /etc/php/7.4/apache2/conf.d/20-mysqli.ini, /etc/php/7.4/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.4/apache2/conf.d/20-pdo_pgsql.ini, /etc/php/7.4/apache2/conf.d/20-posix.ini, /etc/php/7.4/apache2/conf.d/20-readline.ini, /etc/php/7.4/apache2/conf.d/20-redis.ini, /etc/php/7.4/apache2/conf.d/20-shmop.ini, /etc/php/7.4/apache2/conf.d/20-simplexml.ini, /etc/php/7.4/apache2/conf.d/20-sockets.ini, /etc/php/7.4/apache2/conf.d/20-sysmsg.ini, /etc/php/7.4/apache2/conf.d/20-sysvsem.ini, /etc/php/7.4/apache2/conf.d/20-sysvshm.ini, /etc/php/7.4/apache2/conf.d/20-tokenizer.ini, /etc/php/7.4/apache2/conf.d/20-xmlreader.ini, /etc/php/7.4/apache2/conf.d/20-xmlwriter.ini, /etc/php/7.4/apache2/conf.d/20-xsl.ini
- PHP API:** 20190902
- PHP Extension:** 20190902
- Zend Extension:** 320190902
- Zend Extension Build:** API320190902.NTS
- PHP Extension Build:** API20190902.NTS
- Debug Build:** no
- Thread Safety:** disabled
- Zend Signal Handling:** enabled
- Zend Memory Manager:** enabled
- Zend Multibyte Support:** provided by mbstring
- IPv6 Support:** enabled
- DTrace Support:** available, disabled
- Registered PHP Streams:** https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar
- Registered Stream Socket Transports:** tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
- Registered Stream Filters:** zlib *, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk, convert.iconv.*

At the bottom, there is a "Configuration" section for the "apache2handler" and a table with the following details:

Apache Version	Apache/2.4.29 (Ubuntu)
Apache API Version	20120211
Server Administrator	webmaster@localhost

Instalando dependências do MISP

Instale algumas dependências do MISP com o comando:

```
# apt-get install curl gcc git gpg-agent make python python3 openssl \  
redis-server sudo vim zip unzip virtualenv libfuzzy-dev sqlite3 \  
moreutils python3-dev python3-pip libxml2-dev libxslt1-dev \  
zlib1g-dev python-setuptools -qy
```

Instalação do MISP

cert.br nic.br egi.br

Objetivos

Instalação do MISP e de seus componentes

Criação e configuração do banco de dados do MISP

Configuração e *hardening* do site do MISP

Habilitar a rotação de *logs*

Instalação do código do MISP

Criação de variáveis de ambiente

O uso das variáveis de ambiente facilita a personalização da instalação do MISP.

Crie as variáveis de ambiente com os comandos:

```
# export PATH_TO_MISP=/var/www/MISP
# export WWW_USER=www-data
# export SUDO_WWW='sudo -H -u www-data'
# export CAKE="/var/www/MISP/app/Console/cake"
```

Verifique se as variáveis foram criadas com o comando:

```
# export
```

Instalação do código do MISP

Criação de diretórios e *download* do código

Crie o diretório onde o MISP será instalado:

```
# mkdir ${PATH_TO_MISP}
# chown $WWW_USER:$WWW_USER ${PATH_TO_MISP}
```

Esses comandos são equivalentes a:

```
# mkdir /var/www/MISP
# chown www-data:www-data /var/www/MISP
```

Faça download do código do MISP com os comandos:

```
# cd ${PATH_TO_MISP}
# $SUDO_WWW git clone https://github.com/MISP/MISP.git ${PATH_TO_MISP}
# $SUDO_WWW git submodule update --init --recursive
# $SUDO_WWW git submodule foreach --recursive git config core.filemode false
# $SUDO_WWW git config core.filemode false
```

Instalação do código do MISP

Criação do *virtualenv*

Crie o *virtualenv* do python com o comando:

```
# $SUDO_WWW virtualenv -p python3 ${PATH_TO_MISP}/venv
```

Crie o diretório de *cache* do pip com o comando:

```
# mkdir /var/www/.cache/
```

```
# chown $WWW_USER:$WWW_USER /var/www/.cache
```

Instalação do código do MISP

Download e instalação de componentes

Faça *download* e instale os componentes do MISP com os comandos:

```
# cd ${PATH_TO_MISP}/app/files/scripts
# $SUDO_WWW git clone https://github.com/CybOXProject/python-cybox.git
# $SUDO_WWW git clone https://github.com/STIXProject/python-stix.git
# $SUDO_WWW git clone https://github.com/MAECProject/python-maec.git
# $SUDO_WWW git clone https://github.com/CybOXProject/mixbox.git
# cd ${PATH_TO_MISP}/app/files/scripts/mixbox
# $SUDO_WWW ${PATH_TO_MISP}/venv/bin/pip install .
# cd ${PATH_TO_MISP}/app/files/scripts/python-cybox
# $SUDO_WWW ${PATH_TO_MISP}/venv/bin/pip install .
# cd ${PATH_TO_MISP}/app/files/scripts/python-stix
# $SUDO_WWW ${PATH_TO_MISP}/venv/bin/pip install .
# cd ${PATH_TO_MISP}/app/files/scripts/python-maec
# $SUDO_WWW ${PATH_TO_MISP}/venv/bin/pip install .
# cd ${PATH_TO_MISP}/cti-python-stix2
# $SUDO_WWW ${PATH_TO_MISP}/venv/bin/pip install .
```

Instalação do código do MISP

Instalação do PyMISP e demais pacotes

Instale o PyMISP com os comandos:

```
# cd ${PATH_TO_MISP}/PyMISP
# $SUDO_WWW ${PATH_TO_MISP}/venv/bin/pip install .
```

Instale os demais pacotes do MISP com os comandos:

```
# $SUDO_WWW ${PATH_TO_MISP}/venv/bin/pip install \
    git+https://github.com/kbandla/pydeep.git
# $SUDO_WWW ${PATH_TO_MISP}/venv/bin/pip install lief
# $SUDO_WWW ${PATH_TO_MISP}/venv/bin/pip install zmq redis
# $SUDO_WWW ${PATH_TO_MISP}/venv/bin/pip install python-magic
# $SUDO_WWW ${PATH_TO_MISP}/venv/bin/pip install plyara
```

Instalação do CakePHP

Instale o CakePHP com os comandos:

```
# cd ${PATH_TO_MISP}/app
# mkdir /var/www/.composer ; sudo chown $WWW_USER:$WWW_USER /var/www/.composer
# $SUDO_WWW php composer.phar install
```

Habilite os módulos para funcionamento do CakePHP com os comandos:

```
# phpenmod redis
# phpenmod gnupg
```

Habilite o funcionamento dos *workers* do CakePHP com o comando:

```
# $SUDO_WWW cp -fa ${PATH_TO_MISP}/INSTALL/setup/config.php \
${PATH_TO_MISP}/app/Plugin/CakeResque/Config/config.php
```

Criação do banco de dados (1/2)

Acesse o MariaDB como root:

```
# mysql -u root -p
```

Crie o banco de dados do MISP e o usuário que terá acesso a este banco com os comandos:

```
MariaDB> CREATE DATABASE misp;
```

```
MariaDB> CREATE USER 'misp_user'@'localhost' IDENTIFIED BY '<MISP_USER-PASSWORD>';
```

```
MariaDB> GRANT USAGE ON *.* to misp_user@localhost;
```

```
MariaDB> GRANT ALL PRIVILEGES on misp.* to 'misp_user'@'localhost';
```

```
MariaDB> FLUSH PRIVILEGES;
```

```
MariaDB> exit
```

Sugestão para gerar a senha do usuário misp_user:

```
# openssl rand -base64 15
```

Criação do banco de dados (2/2)

Importe o esquema do banco de dados do MISP:

```
# ${SUDO_WWW} cat ${PATH_TO_MISP}/INSTALL/MYSQL.sql | mysql -u misp_user misp -p
```

Confira se a importação foi executada corretamente listando as tabelas do banco de dados misp:

```
# mysql -u misp_user -A misp -p  
MariaDB> show tables;
```

```
Tables_in_misp  
-----  
admin_settings  
attribute_tags  
attributes  
bruteforces  
cake_sessions  
correlations  
decaying_model_mappings  
decaying_models  
event_blacklists  
event_delegations  
event_graph  
event_locks  
event_tags  
events  
favourite_tags  
feeds  
fuzzy_correlate_ssdeep  
galaxies  
galaxy_clusters  
galaxy_elements  
galaxy_reference  
jobs  
logs  
news  
noticelist_entries  
noticelists  
notification_logs  
object_references  
object_relationships  
object_template_elements  
object_templates  
objects  
org_blacklists  
organisations  
posts  
regexp  
rest_client_histories  
roles  
servers  
shadow_attribute_correlations  
shadow_attributes  
sharing_group_orgs  
sharing_group_servers  
sharing_groups  
sightings  
tag_collection_tags  
tag_collections  
tags  
tasks  
taxonomies  
taxonomy_entries  
taxonomy_predicates  
template_element_attributes  
template_element_files  
template_element_texts  
template_elements  
template_tags  
templates  
threads  
threat_levels  
user_settings  
users  
warninglist_entries  
warninglist_types  
warninglists  
whitelist  
-----  
66 rows in set (0.00 sec)
```

Correção das permissões

Para garantir que nenhuma permissão foi definida de forma errada, corrija todas as permissões do diretório de instalação do MISP com os comandos:

```
# chown -R ${WWW_USER}:${WWW_USER} ${PATH_TO_MISP}
# chmod -R 750 ${PATH_TO_MISP}
# chmod -R g+ws ${PATH_TO_MISP}/app/tmp
# chmod -R g+ws ${PATH_TO_MISP}/app/files
# chmod -R g+ws ${PATH_TO_MISP}/app/files/scripts/tmp
```

Configuração do *site* do MISP (1/2)

Crie o arquivo `/etc/apache2/sites-available/misp-ssl.conf` seguindo a sugestão de configuração *Intermediate* do site Mozilla SSL Configuration Generator :

```
<VirtualHost *:80>
    ServerName <seu FQDN>

    Redirect permanent / https://<seu FQDN>

    LogLevel warn
    ErrorLog /var/log/apache2/misp.local_error.log
    CustomLog /var/log/apache2/misp.local_access.log combined
    ServerSignature Off
</VirtualHost>

<VirtualHost *:443>
    ServerAdmin admin@<seu FQDN>
    ServerName <seu FQDN>
    DocumentRoot /var/www/MISP/app/webroot
    <Directory /var/www/MISP/app/webroot>
        Options -Indexes
        AllowOverride all
        Order allow,deny
        allow from all
    </Directory>
```

[CONTINUA NO PRÓXIMO SLIDE]

Baseado em: Mozilla SSL Configuration Generator (<https://ssl-config.mozilla.org>)

Configuração do *site* do MISP (2/2)

[CONTINUAÇÃO DO *SLIDE* ANTERIOR]

```
SSLEngine On
SSLCertificateFile /etc/letsencrypt/live/<seu FQDN>/fullchain.pem
SSLCertificateKeyFile /etc/letsencrypt/live/<seu FQDN>/privkey.pem
```

```
LogLevel warn
ErrorLog /var/log/apache2/misp.local_error.log
CustomLog /var/log/apache2/misp.local_access.log combined
ServerSignature Off
Header always set Strict-Transport-Security "max-age=63072000"
```

```
</VirtualHost>
```

```
ServerTokens Prod
SSLProtocol all -SSLv3 -TLSv1 -TLSv1.1
SSLCipherSuite ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-
AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-
CHACHA20-POLY1305:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384
SSLHonorCipherOrder off
SSLSessionTickets off
```

```
SSLUseStapling On
SSLStaplingCache "shmcb:logs/ssl_stapling(32768)"
```

Habilitando o *site* do MISP

Verifique se o arquivo de configuração criado está correto com o comando:

```
# apache2ctl configtest
```

Desabilite o módulo `status` do Apache com o comando:

```
# a2dismod status
```

Habilite os módulos `ssl`, `rewrite` e `headers` com os comandos:

```
# a2enmod ssl
```

```
# a2enmod rewrite
```

```
# a2enmod headers
```

Desabilite o *site* padrão do Apache e habilite o *site* do MISP com os comandos:

```
# a2dissite 000-default
```

```
# a2ensite misp-ssl
```

Reinicie o Apache com o comando:

```
# systemctl restart apache2
```

Rotação de *Logs*

Habilite a rotação de *logs* do MISP com os comandos:

```
# cp ${PATH_TO_MISP}/INSTALL/misp.logrotate /etc/logrotate.d/misp  
# chmod 0640 /etc/logrotate.d/misp
```

Configuração do MISP

cert.br nic.br egi.br

Objetivos

Criar os arquivos de configuração para o funcionamento do MISP

Habilitar os *workers*

Realizar configurações iniciais e personalizações no MISP

Criando os arquivos de configuração

Copie os arquivos *default* fornecidos pelo MISP para criar os arquivos de configuração

```
# $SUDO_WWW cp -a ${PATH_TO_MISP}/app/Config/bootstrap.default.php \  
    ${PATH_TO_MISP}/app/Config/bootstrap.php  
  
# $SUDO_WWW cp -a ${PATH_TO_MISP}/app/Config/database.default.php \  
    ${PATH_TO_MISP}/app/Config/database.php  
  
# $SUDO_WWW cp -a ${PATH_TO_MISP}/app/Config/core.default.php \  
    ${PATH_TO_MISP}/app/Config/core.php  
  
# $SUDO_WWW cp -a ${PATH_TO_MISP}/app/Config/config.default.php \  
    ${PATH_TO_MISP}/app/Config/config.php
```

Configurando o acesso ao banco de dados

Edite o arquivo `$PATH_TO_MISP/app/Config/database.php`

```
# $SUDO_WWW vi $PATH_TO_MISP/app/Config/database.php
```

Encontre e altere as seguintes linhas:

```
'login' => 'misp_user',  
'password' => '<MISP_USER-PASSWORD>',  
'database' => 'misp',
```

Configurando o *salt*

É importante gerar um novo *salt* antes de alterar a senha do usuário administrador do MISP

Gere um novo *salt* com o comando:

```
# openssl rand -base64 24
```

Edite o arquivo `config.php`:

```
# $SUDO_WWW vi $PATH_TO_MISP/app/Config/config.php
```

Localize a linha `'salt'` => `' '` e coloque dentro das aspas a *string* gerada pelo comando `openssl`

Corrigindo as permissões

Para garantir que as permissões dos arquivos de configuração do MISP estão corretas, faça a correção delas com os comandos:

```
# chown -R $WWW_USER:$WWW_USER ${PATH_TO_MISP}/app/Config  
# chmod -R 750 ${PATH_TO_MISP}/app/Config
```

Habilitando os *workers* (1/2)

Habilite a permissão de execução no *script* que inicia os *workers*

```
# chmod +x $PATH_TO_MISP/app/Console/worker/start.sh
```

Crie no diretório `/etc/systemd/system/` o arquivo `misp-workers.service` com o seguinte conteúdo:

```
[Unit]
Description=MISP background workers
After=network.target

[Service]
Type=forking
User=www-data
Group=www-data
ExecStart=/var/www/MISP/app/Console/worker/start.sh
Restart=always
RestartSec=10

[Install]
WantedBy=multi-user.target
```

Habilitando os *workers* (2/2)

Habilite e inicie o *daemon* recém criado no sistema com os comandos:

```
# systemctl daemon-reload  
# systemctl enable --now misp-workers
```

Configurações iniciais do MISP

Atualizando o banco de dados e definindo o *path* para o *virtualenv*

Atualize o banco de dados do MISP com os comandos

```
# $SUDO_WWW -- $CAKE userInit -q  
# $SUDO_WWW -- $CAKE Admin runUpdates
```

Defina o *path* para o *virtualenv* com o comando:

```
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.python_bin" \  
"${PATH_TO_MISP}/venv/bin/python"
```

Configurações iniciais do MISP

Configurando *timeouts* e a URL do MISP

Defina os timeouts do MISP com os comandos:

```
# $SUDO_WWW -- $CAKE Admin setSetting "Session.autoRegenerate" 0
# $SUDO_WWW -- $CAKE Admin setSetting "Session.timeout" 600
# $SUDO_WWW -- $CAKE Admin setSetting "Session.cookieTimeout" 3600
```

Defina a URL do MISP com os comandos:

```
# $SUDO_WWW -- $CAKE Baseurl https://<seu FQDN>
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.external_baseurl" https://<seu FQDN>
```

Configurações iniciais do MISP

Configurando a organização *default* e outras variáveis

Configure a organização *default* e os endereços de *e-mail* do MISP com os comandos:

```
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.host_org_id" 1
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.email" "<EMAIL>"
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.disable_emailing" true
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.contact" "<EMAIL>"
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.disable_restalert" true
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.showCorrelationsOnIndex" true
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.default_event_tag_collection" 0
```

Configurações iniciais do MISP

Configurando *plug-ins* (1/3)

Defina as configurações *default* do cortex com os comandos:

```
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.Cortex_services_enable" false
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.Cortex_services_url"
"http://127.0.0.1"
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.Cortex_services_port" 9000
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.Cortex_timeout" 120
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.Cortex_authkey" ""
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.Cortex_ssl_verify_peer" false
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.Cortex_ssl_verify_host" false
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.Cortex_ssl_allow_self_signed" true
```

Configurações iniciais do MISP

Configurando *plug-ins* (2/3)

Defina as configurações *default* do sightings com os comandos:

```
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.Sightings_policy" 0
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.Sightings_anonymise" false
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.Sightings_range" 365
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.Sightings_sighting_db_enable" false
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.CustomAuth_disable_logout" false
```

Configurações iniciais do MISP

Configurando *plug-ins* (3/3)

Defina as configurações *default* do RPZ com os comandos:

```
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.RPZ_policy" "DROP"
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.RPZ_walled_garden" "127.0.0.1"
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.RPZ_serial" "\$date00"
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.RPZ_refresh" "2h"
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.RPZ_retry" "30m"
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.RPZ_expiry" "30d"
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.RPZ_minimum_ttl" "1h"
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.RPZ_ttl" "1w"
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.RPZ_ns" "localhost."
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.RPZ_ns_alt" ""
# $SUDO_WWW -- $CAKE Admin setSetting "Plugin.RPZ_email" "root.localhost"
```

Configurações iniciais do MISP

Configurando o Redis

Configure o Redis com os comandos

```
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.redis_host" "127.0.0.1"  
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.redis_port" 6379  
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.redis_database" 13  
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.redis_password" ""
```

Configurações iniciais do MISP

Gerando *logs* dos endereços IP de clientes

Configure o MISP para gerar *logs* dos endereços IPs dos usuários com os comandos:

```
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.log_client_ip" true  
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.log_auth" true
```

Configurações iniciais do MISP

Personalizações

Personalize o MISP com os comandos:

```
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.footermidleft" ""  
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.footermidright" "Operated by <SUA_ORG>"  
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.welcome_text_top" "<SUA_ORG>"  
# $SUDO_WWW -- $CAKE Admin setSetting "MISP.welcome_text_bottom" ""
```

Configurações iniciais do MISP

Atualizações

Atualize galáxias, taxonomias e outros objetos com os comandos:

```
# $SUDO_WWW -- $CAKE Admin updateGalaxies  
# $SUDO_WWW -- $CAKE Admin updateTaxonomies  
# $SUDO_WWW -- $CAKE Admin updateWarningLists  
# $SUDO_WWW -- $CAKE Admin updateNoticeLists  
# $SUDO_WWW -- $CAKE Admin updateObjectTemplates "1337"
```

Configurações iniciais do MISP

Outras configurações

Para mais configurações *default* do MISP, acesse: <https://cert.br/misp/tutorial-ubuntu/>

Uso do MISP via interface Web

cert.br nic.br egi.br

Objetivos

Fazer o primeiro *login* na instância e alterar a senha do usuário administrador

Alterar o usuário administrador *default*

Alterar a organização *default*

Criar novos usuários

Criar servidores de sincronia

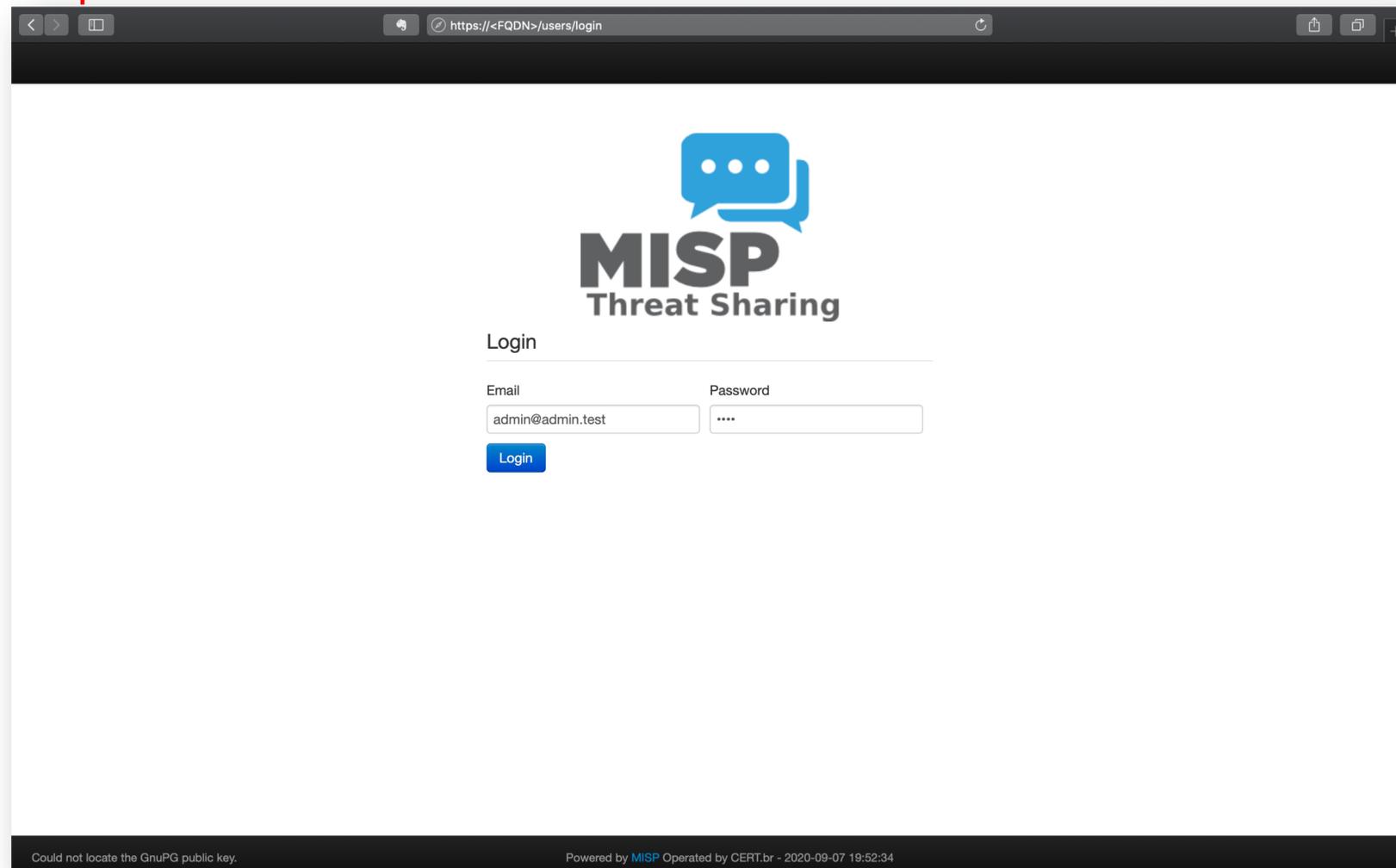
Atualizar o MISP

Primeiro *login* (1/2)

Acesse o servidor MISP pela URL: <https://<FQDN>>

Acesse o MISP com os dados:

- Email: admin@admin.test
- Password: test



Primeiro *login* (2/2)

O MISP exige a alteração da senha do usuário `admin` após o primeiro *login*

Change Password

Password ⓘ

Confirm Password

Utilize uma senha forte.

Alteração do usuário admin (1/2)

É recomendado alterar o *e-mail* do usuário `admin@admin.test`

Para isso clique em “Administration – List Users”

Na página “Users index” localize o usuário `admin@admin.test` e clique em *edit*

Users index

Click [here](#) to reset the API keys of all sync and org admin users in one shot. This will also automatically inform them of their new API keys.

« previous next »



Id	Org	Role	Email	authkey	Autoalert	Contactalert	PGP Key	NIDS SID	Terms Accepted	Last Login	Created	Disabled	Actions
1	ORGNOME	admin	admin@admin.test	*****	x	x	x	4000000	x	2020-09-07 20:03:36		x	   

Page 1 of 1, showing 1 records out of 1 total, starting on record 1, ending on 1

« previous next »



Alteração do usuário admin (2/2)

Na janela “Admin Edit User” altere o *e-mail* do usuário admin

Após alterar o *e-mail*, clique em *submit*

Admin Edit User

Email
workshop-admin@cert.br 

Set password [Reset Auth Key](#)

Organisation: ORGNAME Role: admin

Authkey: laZxsdxOlod4m3JljO8nJcGcB43 Nids Sid: 4000000

Sync user for: Not bound to a server

GnuPG key
Paste the user's GnuPG key here or try to retrieve it from the CIRCL key server by clicking on "Fetch GnuPG key" below.

Terms accepted Change Password Receive alerts when events are published Receive alerts from "contact reporter" requests

Disable this user account

Alterando a organização inicial (1/4)

Altere o nome da organização ORGNAME para um nome que reflita sua organização.

Para isso clique em “Administration → List Organisations”

Na Janela “Local organisations having a presence on this instance” identifique a organização ORGNAME e clique em editar

Local organisations having a presence on this instance

« previous next » [View all](#)

Local organisations													Known remote organisations		All organisations	
Enter value to search													Filter			
Id	Logo	Name	UUID	Description	Nationality	Sector	Type	Contacts	Added by	Local	Users	Restrictions	Actions			
1	ORGNAME	ORGNAME	b6f8983e-8735-41f2-a74e-5978ace79b07	Automatically generated admin organisation	Not specified		ADMIN		Unknown	Yes	1		✎ 🗑 👁			

Page 1 of 1, showing 1 records out of 1 total, starting on record 1, ending on 1

« previous next » [View all](#)



Alterando a organização inicial (2/4)

Edit Organisation

If the organisation should have access to this instance, make sure that the Local organisation setting is checked.
If you would only like to add a known external organisation for inclusion in sharing groups, uncheck the Local organisation setting.

Local organisation

Mandatory fields. Leave the UUID field empty if the organisation doesn't have a UUID from another instance.

Organisation Identifier

UUID

A brief description of the organisation

Bind user accounts to domains (line separated)

The following fields are all optional.

Logo (48x48 png)
 nenhum arquivo selecionado

Nationality Sector

Type of organisation

Contacts

Alterando a organização inicial (3/4)

Na janela “Edit Organisation” preencha os seguintes dados:

- Marque o *checkbox* identificando a organização como local
- Em “Organisation Identifier”:
 - Substitua o nome ORGNAME pelo nome da sua organização
- Em UUID:
 - Mantenha a *string* gerada pelo sistema ou substitua pela *string* da sua organização caso já exista
- Em “A brief description of the organisation”:
 - Coloque uma breve descrição da sua organização
- Em “Bind user accounts to domains (line separated)”:
 - Você pode colocar uma lista de domínios permitidos para a criação dos usuários (contas de *e-mail*)

Alterando a organização inicial (4/4)

- Também é possível inserir as seguintes características de uma organização:
 - Logo
 - Nacionalidade
 - Setor
 - Tipo
 - Detalhes de contato

Criando usuários (1/4)

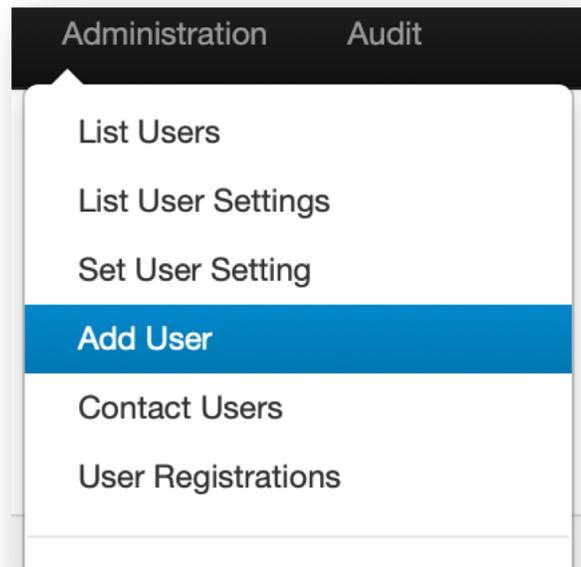
O usuário com privilégios de administrar o MISP deve ser utilizado apenas para tarefas administrativas, como por exemplo:

- Criar novos usuários
- Criar novas organizações
- Criar servidores de sincronia
- Atualizar o sistema
- Outras atividades administrativas

Para utilização diária, é recomendado utilizar um usuário sem privilégios administrativos.

Criando usuários (2/4)

Para criar um novo usuário, clique em “Administration → Add User”



Criando usuários (3/4)

Na janela “Admin Add User”, preencha os seguintes campos:

- Email
 - Endereço de *e-mail* do usuário
- Marque a opção “Set password”
 - Digite e confirme a senha do usuário
- Organisation
 - Escolha a organização deste usuário
- Role
 - Escolha o papel do usuário
- Authkey
 - Gerada automaticamente pelo sistema
- Nids Sid
 - Utilizado pelo IDS

The screenshot shows the 'Admin Add User' form with the following fields and options:

- Email:** usuario@cert.br
- Set password:**
- Password:** [Redacted]
- Confirm Password:** [Redacted]
- Organisation:** CERT.br Workshop-MISP
- Role:** User
- Authkey:** Xd70lySsBTKS1xUlnmmtolQz6B.
- Nids Sid:** [Empty]
- GnuPG key:** [Text area with instructions: "Paste the user's GnuPG key here or try to retrieve it from the CIRCL key server by clicking on 'Fetch GnuPG key' below."]
- Fetch GnuPG key:** [Button]
- Receive alerts when events are published:**
- Receive alerts from "contact reporter" requests:**
- Disable this user account:**
- Send credentials automatically:**
- Submit:** [Button]

Criando usuários (4/4)

- GnuPG key
 - Caso esteja utilizando chaves PGP, digite a chave do usuário ou clique no botão "Fetch GnuPG key"
- Receive alerts when events are published
 - Se selecionada, esta opção colocará o usuário em uma lista onde ele vai receber *e-mails* para cada evento publicado
- Receive alerts from "contact reporter" requests
 - Se selecionada, esta opção colocará o usuário em uma lista onde ele vai receber *e-mails* sempre que outro usuário tentar entrar em contato reportando eventos daquela organização
- Disable this user account
 - Se selecionada, esta opção desabilita a conta do usuário.
 - Os desenvolvedores do MISP recomendam utilizar esta opção ao invés de apagar um usuário
- Send credentials automatically
 - Se selecionada, esta opção enviará as credenciais do usuário.

Papéis (*roles*) dos usuários

- Admin
 - Privilégios para administrar o sistema
- Org admin
 - Privilégios para administrar a organização
- User
 - Podem criar eventos mas não publicá-los
- Publisher
 - Criam e publicam eventos dentro de uma organização
- Sync User
 - Utilizado para sincronia com outras instâncias MISP
- Read Only
 - Podem apenas ler eventos dentro de uma organização

Distribuição de eventos

Os desenvolvedores do MISP recomendam que a sincronia entre servidores deve ser feita apenas via *push* para evitar problemas de transferência de dados.

Para enviar eventos via *push* para uma organização parceira, é necessário criar um servidor de sincronia para esta organização

Para receber eventos de uma organização parceira, é necessário que esta organização crie um servidor de sincronia para sua organização

Configuração da instância receptora de eventos

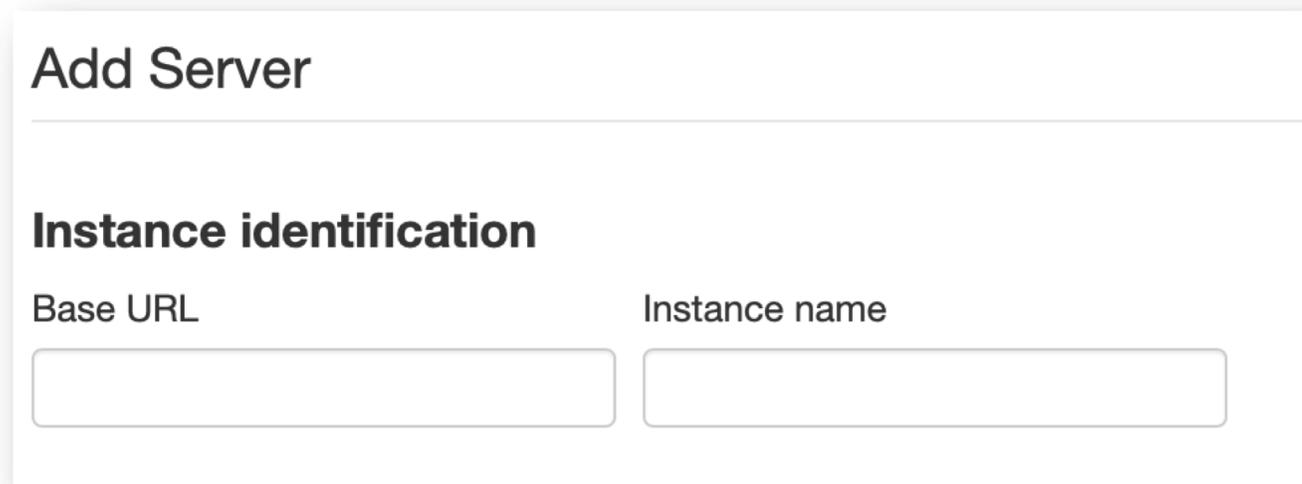
Crie na instância que receberá eventos:

- Uma organização local com os mesmos nome e UUID da organização emissora dos eventos
- Crie nesta organização um usuário do tipo “sync-user”
- Envie para os administradores da instância emissora os seguintes dados:
 - URL da instância receptora dos eventos
 - Organisation Identifier da organização receptora
 - UUID da organização que receberá eventos
 - Authkey do usuário “sync-user”

Configuração da instância emissora de eventos (1/3)

Na instância emissora dos eventos, crie um servidor de sincronia:

- Na tela principal do MISP, clique em “Sync Actions → List Servers”
- Na tela “Servers”, localize o link “New Servers”
- Na tela “Add Server” entre com a seguinte informação:
 - Em Instance Identification:
 - Base URL: URL da instância receptora
 - Instance Name: Nome da organização receptora



Add Server

Instance identification

Base URL

Instance name

[CONTINUA NO PRÓXIMO SLIDE]

Configuração da instância emissora de eventos (2/3)

[CONTINUAÇÃO DO SLIDE ANTERIOR]

- Em “Instance ownership and credentials”:

- Organisation Type: Selecione o tipo de organização, geralmente “New external organisation”
- Remote Organisation's Name: Preencha com o valor “Organisation Identifier” recebido da organização receptora
- Remote Organisation's UUID: Preencha com o valor “UUID” recebido da organização receptora
- Authkey: Preencha com a authkey do sync-user criado na instância receptora

Instance ownership and credentials

Information about the organisation that will receive the events, typically the remote instance's host organisation.

Organisation Type	Remote Organisation's Name	Remote Organisation's UUID
New external organisation	<input type="text"/>	<input type="text"/>

Ask the owner of the remote instance for a sync account on their instance, log into their MISP using the sync user's credentials and retrieve your API key by navigating to Global actions -> My profile. This key is used to authenticate with the remote instance.

Authkey

[CONTINUA NO PRÓXIMO SLIDE]

Configuração da instância emissora de eventos (3/3)

[CONTINUAÇÃO DO SLIDE ANTERIOR]

- Em “Enabled synchronisation methods”:
 - Marque a opção “Push”
- Ao final clique em Submit

Enabled synchronisation methods

Push Pull Push Sightings Caching Enabled

Misc settings

Unpublish Event
 Publish Without Email
 Self Signed
 Skip proxy (if applicable)

Server certificate file (*.pem): **Not set.**
[Add certificate file](#) [Remove certificate file](#)

Client certificate file: **Not set.**
[Add certificate file](#) [Remove certificate file](#)

Push rules:
[Modify](#)

Pull rules:
[Modify](#)

[Submit](#)

Formas de distribuição de eventos

Os eventos podem ser distribuídos da seguinte forma:

- Your organisation only
 - Apenas usuários da sua organização recebem os eventos
- This community only
 - Usuários de outras organizações no seu servidor MISP recebem os eventos
- Connected communities
 - Usuários de organizações de servidores MISP conectados diretamente ao seu servidor MISP recebem os eventos
- All communities
 - Todos os usuários recebem os eventos
- Sharing group
 - Apenas organizações selecionadas em servidores selecionados recebem os eventos

Atualização do MISP (1/2)

É importante manter o código do MISP sempre atualizado.

- Para verificar se existem atualizações disponíveis, na janela principal do MISP clique em “Administration → Server Settings & Maintenance”
- Na tela “Server Settings & Maintenance” clique em “Diagnostics”
- Verifique se existem atualizações disponíveis ou se o MISP está atualizado

Server Settings & Maintenance

Overview MISP settings (40 ) Encryption settings (8 ) Proxy settings (5) Security settings (3) Plugin settings (87 ) **Diagnostics (6)** Manage files  Workers 

MISP version

Every version of MISP includes a json file with the current version. This is checked against the latest tag on github, if there is a version mismatch the tool will warn you about it. Make sure that you update MISP regularly.

Currently installed version... **v2.4.130** (6635471b8772fa84c409c9764839e7bbd2bfbde5)

Latest available version... **v2.4.131** (d06cd0d3e9c203fae88b9d8f7a3a10280813f9ec)

Status... **Outdated version**

Current branch... **2.4**

Update MISP  [View Update Progress](#)

Atualização do MISP (2/2)

- Caso necessite de atualizações, clique no botão “Update MISP”
- Aguarde até a atualização terminar
- Recarregue a página e verifique se o MISP foi atualizado

Server Settings & Maintenance

Overview MISP settings (40 ) Encryption settings (8 ) Proxy settings (5) Security settings (3) Plugin settings (87 ) **Diagnostics (5)** Manage files Workers 

MISP version

Every version of MISP includes a json file with the current version. This is checked against the latest tag on github, if there is a version mismatch the tool will warn you about it. Make sure that you update MISP regularly.

Currently installed version... **v2.4.131** (d06cd0d3e9c203fae88b9d8f7a3a10280813f9ec)

Latest available version... **v2.4.131** (9cf42988fb40909391e5a71fde97ac14b7cb76f0)

Status... **OK**

Current branch... **2.4**

Update MISP

 [View Update Progress](#)

Uso do MISP de maneira automatizada

cert.br nic.br egi.br

REST API

O MISP tem uma REST API que possibilita:

- gerenciar (adicionar, atualizar e remover) eventos
- gerenciar atributos
- gerenciar *tags*
- gerenciar organizações
- gerenciar usuários
- submeter *sightings*
- obter estatísticas de atributos/*tags*
- etc

Referências: <https://www.circl.lu/doc/misp/automation/>

Exemplo de consulta utilizando curl

Buscando eventos publicados no último dia:

```
curl \  
-d '{"returnFormat":"json","publish_timestamp":"1d"}' \  
-H "Authorization: authkey" \  
-H "Accept: application/json" \  
-H "Content-type: application/json" \  
-X POST https://<FQDN>/events/restSearch
```

PyMISP

O PyMISP é uma biblioteca do Python utilizada para acessar o MISP através da sua REST API.

O PyMISP permite buscar eventos, adicionar ou atualizar eventos e/ou atributos, adicionar ou atualizar binários ou pesquisar por atributos.

Para utilizar o PyMISP é necessário ter uma *auth key* em uma instância MISP.

Referências:

- <https://www.circl.lu/doc/misp/pymisp/>
- <https://pymisp.readthedocs.io/en/latest/>

PyMISP – Capacidades

Com o PyMISP é possível:

- Adicionar, buscar, atualizar, publicar e apagar eventos
- Adicionar ou remover *tags*
- Adicionar atributos de arquivos: *hashes, registry key, patterns, pipe, mutex*
- Adicionar atributos de redes: Endereços IP de origem e destino, *hostname*, domínios, URL, entre outros
- Adicionar atributos de e-mail: Origem, destino, assunto, anexos, entre outros
- Fazer *upload/download* de binários
- Atualizar *sightings*
- Pesquisar por palavras chaves e por atributos

Instalação do PyMISP

É possível instalar o PyMISP pelo utilitário `pip` ou diretamente do seu repositório no GitHub.

Para instalar o PyMISP pelo `pip`, digite o comando:

```
pip install pymisp
```

Para instalar o PyMISP pelo GitHub, digite os comandos:

```
git clone https://github.com/MISP/PyMISP.git && cd PyMISP  
python setup.py install
```

Arquivo *keys.py*

Uma convenção para facilitar o gerenciamento de múltiplos *scripts*

```
#!/usr/bin/env python3

misp_url = 'https://<FQDN>'
misp_key = '<AUTHKEY>' # MISP auth key found on the MISP web interface
misp_verifycert = True
```

Exemplo de código (1/4)

```
#!/usr/bin/env python3

from pymisp import ExpandedPyMISP, MISPEvent, MISPAttribute
from keys import misp_url, misp_key, misp_verifycert

if __name__ == '__main__':
    # connect to misp instance
    misp = ExpandedPyMISP(misp_url, misp_key, misp_verifycert)
```

[CONTINUA NO PRÓXIMO SLIDE]

Exemplo de código (2/4)

[CONTINUAÇÃO DO SLIDE ANTERIOR]

```
# create event
my_event = MISPEvent()

my_event.info = 'Phishing: www.example.org'

# threat IDs: 1 = High / 2 = Medium / 3 = Low / 4 = Undefined
my_event.threat_level_id = 1
# analysis IDs: 0 = Initial / 1 = Ongoing / 2 = Completed
my_event.analysis = 1
# distribution IDs
# 0 = Your Organization only / 1 = This community only / 2 = Connected communities /
# 3 = All communities
my_event.distribution = 1

# add basic information to event
my_event = misp.add_event(event=my_event, pythonify=True)
```

[CONTINUA NO PRÓXIMO SLIDE]

Add Event

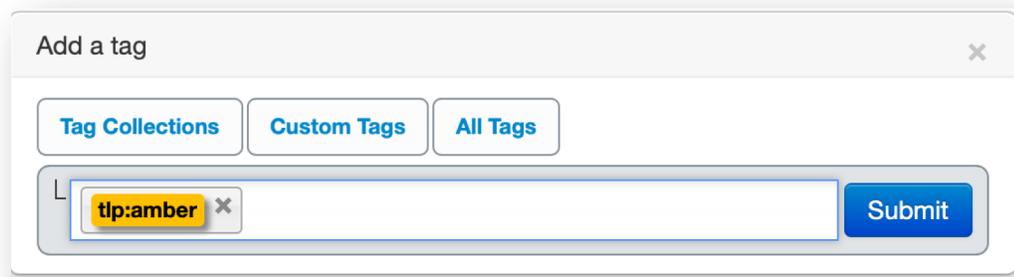
Date	Distribution ⓘ
<input type="text" value="2020-09-11"/>	<input type="text" value="This community only"/>
Threat Level ⓘ	Analysis ⓘ
<input type="text" value="High"/>	<input type="text" value="Ongoing"/>
Event Info	
<input type="text" value="Phishing: www.example.org"/>	
Extends Event	
<input type="text" value="Event UUID or ID. Leave blank if not applicable."/>	
<input type="button" value="Submit"/>	

Exemplo de código (3/4)

[CONTINUAÇÃO DO SLIDE ANTERIOR]

```
# add event tag
# tag IDs: 1 = tlp:red / 2 = tlp:amber / 3 = tlp:green / 4 = tlp:white
misp.tag(my_event, "2")

# the phishing URL
phishing_url = 'http://www.example.org/phishing_page.html'
```



Id ↓	Exportable	Hidden	Name
1	✓	✗	tlp:red
2	✓	✗	tlp:amber
3	✓	✗	tlp:green
4	✓	✗	tlp:white

[CONTINUA NO PRÓXIMO SLIDE]

Exemplo de código (4/4)

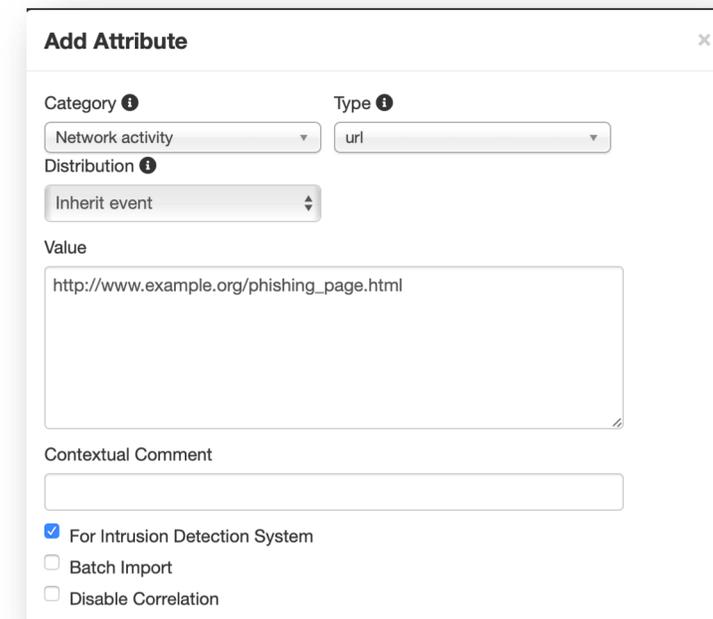
[CONTINUAÇÃO DO SLIDE ANTERIOR]

```
# add event attribute
url_Attribute = MISPAtribute()
url_Attribute.category = "Network activity"
url_Attribute.type = "url"
url_Attribute.value = phishing_url
url_Attribute.comment = "Phishing URL"
url_Attribute.to_ids = True
misp.add_attribute(my_event.id, attribute=url_Attribute, pythonify=True)

# publish event
misp.publish(event=my_event.id)

print(my_event.to_json())

# EOF
```



Add Attribute [X]

Category ⓘ Network activity ▾ Type ⓘ url ▾

Distribution ⓘ Inherit event ▾

Value

http://www.example.org/phishing_page.html

Contextual Comment

For Intrusion Detection System

Batch Import

Disable Correlation

Recomendações para automação

Ter um usuário específico para essa finalidade.

Começar “pequeno”

Obrigado

✉️ marcus@cert.br

✉️ Notificações para: cert@cert.br

📧 [@certbr](https://twitter.com/certbr)

<https://cert.br/>

nic.br **egi.br**

www.nic.br | www.cgi.br