



SIMETBox

Informações de integração do firmware

draft v3, 2024-10-04

Henrique de Moraes Holschuh <henrique@nic.br>

Sumário

Sumário	2
Alterações	4
TODO	5
Documentação para Integradores	5
Introdução	6
Sistema operacional	8
Alterações necessárias no OpenWRT	8
BUSYBOX	10
CONFIG do BUSYBOX (OpenWRT 18.06+):	10
CONFIG adicional do BUSYBOX (OpenWRT 15.05)	10
LIBCURL e CURL, TLS	11
LIBCURL config (OpenWRT):	11
Certificate store (pacote ca-bundle)	11
Motor de criptografia (mbedtls, wolfssl, openssl)	12
Motor de Medição SIMET2	13
Licenciamento	13
Embarque do motor de medição em equipamentos	13
Atualizações de segurança	14
Equipamentos em fim de vida útil (End Of Life)	14
Geolocalização	14
Integração com o Sistema Operacional	16
Privilégios root e o motor de medição SIMET2	16
Pacotes OpenWRT do motor de medição	17
Outros pacotes do Sistema Operacional	18
Código fonte	19
Preparação de ambiente,e recomendações	19
Ambiente de build e desenvolvimento	19
Obtendo o código fonte	20
Preparação do código fonte	20
exemplo de openwrt-openwrt/feeds.conf :	21
Motor de medição:	22
simet-ma	23
Idéia geral de como a integração do sistema operacional com o simet-ma funciona:	24
Empacotamento e build openwrt do motor de medição	25

Pacotes a ignorar	25
Tour do “empacotamento” para openwrt do motor de medição	26
simetbox-openwrt-simet-lmapd	26
simetbox-openwrt-simet-ma	27
OpenWRT config	29
Estado dos símbolos de configuração (openwrt-19.07):	29

Alterações

draft v2 (2021-11-16):

- Adicionadas informações referentes ao licenciamento do motor de medição (ver seção “Licenciamento” no capítulo sobre o Motor de Medição SIMET2).
- SIMET1 agora é desabilitado por padrão pelo config de `simetbox-openwrt-feed/simetbox-openwrt-base`
- UBUS: `ubus call simet_ma results_credentials` foi ampliada para retornar uma URL, que deve ser utilizada (por exemplo, em um IFRAME) pela UI do medidor para permitir o acesso do usuário às medições realizadas.

A UI padrão (extensão ao “LuCI” do OpenWRT) foi atualizada para utilizar esta chamada, e pode ser utilizada como exemplo.

É necessário que o medidor já tenha sido registrado, e que tenha obtido os *tokens de identificação* (procedimento realizado durante o *boot* e a cada 24H). Essa `ubus call` retorna mais informações, que devem ser ignoradas por *firmwares* de terceiros: use apenas a URL.

draft v3 (2024-10-04):

- Parcialmente revisado para versões mais recentes do motor de medição e empacotamento OpenWRT
- TODO atualizado
- Mencionada a documentação para integradores, que é mais atual que este documento, e tem abrangência maior.

TODO

- Documentar a parte relevante da API UBUS `simet_ma` para integração com *firmware* de terceiros
- Recomendamos **veementemente** que o *firmware* de terceiros integre com o motor de medição via API UBUS, e *initscripts* e *cron scripts*, sem mais nenhuma superfície de contato. É melhor para manutenção futura, irá facilitar atualizações do motor de medição.
- Documentar a identificação do firmware de terceiro, *importante e obrigatória*, (**SIMETLIBDIR/conf.d/firmware-vendor.conf**) – consta na documentação de integração.
- Documentar que mantemos registro de `uci system.system[0].hostname` e de `uci system.system[0].description` que consideramos uma extensão útil e desejável (porém inteiramente opcional) para o firmware de terceiro permitir a configuração. Note que está presente no OpenWRT 19.07+ e que fizemos o backport disso (LuCI).
- Documentação do “virtual label” – consta na documentação de integração.

Documentação para Integradores

Este documento específico para o OpenWRT / SIMETBox é complementar à documentação de integração do motor de medição SIMET, presente em:

<https://docs.medicoes.nic.br/simet-integrado/>

É necessário ler ambas as documentações para uma integração com sucesso.

Eventualmente, este documento será convertido para o formato *web* e passará a fazer parte da documentação disponível em `docs.medicoes.nic.br`.

Introdução

O firmware SIMETBox é utilizado pelo NIC.br para distribuir seu medidor de qualidade de conexão à Internet, no formato “roteador doméstico” ou “pequeno medidor embarcado”.

Lista geral de funcionalidades do SIMETBox:

- Herdadas do OpenWRT:
 - Modular, escolha de funcionalidade de acordo com a necessidade
 - IPv6 completo (inclusive DHCPv6-PD recursivo, 464XLAT, etc)
 - Suporte a USB (*storage*, modem 3G/4G, etc)
 - CPE completa (roteamento, bridge cabeada e *wireless*, VLANs, etc)
 - Gerenciamento remoto (SSH, web, Zabbix, SNMP, TR-069 simples)
 - Servidor local NTP, DNS, DHCP/v6, etc.
- SIMET: Medições principais:
 - Vazão (velocidade) TCP IPv4, IPv6
 - Latência, jitter, perda de pacotes, reordenamento de pacotes(*), IPv4/IPv6
 - Disponibilidade de conexão à Internet [para servidores remotos], IPv4/IPv6
 - Rota ativa (via *traceroute*) até o servidor de medição
- SIMET: Medições auxiliares (e opcionais):
 - Medições de qualidade DNS, DNSSEC
 - Detecção de interceptação de DNS
 - Detecção de interceptação de site web https
 - Mapeamento de conectividade aos pontos de troca de tráfego
 - *traceroute*, *ping* para endereços específicos de interesse
- NIC.br: extensões ao OpenWRT (disponíveis como software livre mediante pedido explícito ao nosso suporte):
 - Ajuste do relógio do roteador em caso de NTP bloqueado (HTP)
 - Sysupgrade mais confiável com pouca RAM, para OpenWRT 15.05
 - Cliente de atualização e *hotfix* OTA para frotas de equipamentos OpenWRT, usando assinaturas criptográficas Ed25519 (lado servidor não disponível, mas simples de implementar).

(*) reordenamento de pacotes é medido, mas não informamos o resultado ao usuário ainda. Não precisará nenhuma alteração no medidor/firmware para informar ao usuário do resultado dessa medição.

O SIMETBox consiste em:

- Motor de Medição SIMET2 “SIMET-MA”
 - Medições em executáveis compilados, código fonte em “C” (*/usr/bin*)
 - Scripts shell (*bash / busybox sh*)
 - *Spool* com os relatórios de medições ainda não enviados
 - Informações de configuração persistente (em */etc/simet*)
 - Informações de estado voláteis (em */var/run/simet, /var/run/simet_publico*)
- Sistema Operacional OpenWRT com atualizações

- Todas as atualizações estão disponíveis no *GitHub* e tipicamente consistem em *backports* de funcionalidade e bugfixes do OpenWRT
- A equipe SIMET submete ao OpenWRT todas as modificações relevantes.
- Integração do motor de medição SIMET-MA com o OpenWRT
 - Initscripts, cron scripts, first-boot config scripts, version-update migration scripts
 - Camada UBUS rpc para JSON-RPC sobre http (para uso por app mobile), e também para uso do LuCI
 - Módulos LuCI para ampliar a interface web interna padrão do OpenWRT e controlar localmente o motor de medição SIMET
- (Apenas firmware oficial NIC.br): módulo de gestão remota e atualização remota
 - Scripts que tornam o “sysupgrade” um pouco mais confiável com pouca RAM livre. Para os serviços rodando no equipamento, desliga rádios e remove os módulos *kernel* referentes ao stack IEEE802.11 da memória caso haja pouca RAM livre durante o sysupgrade
 - Script que abre túnel SSH reverso para para suporte remoto de emergência
 - Script que faz *download* de imagem de atualização *sysupgrade*, verifica assinatura criptográfica Ed25519, e aplica via *sysupgrade*. Também permite ajustes emergenciais no *firmware* instalado, configuração, ou executar um *script* (sempre verificando assinatura criptográfica antes de aceitar o comando).

Para efeito de integração no *firmware* de terceiros, o “módulo de gestão remota e atualização remota” **não é relevante**. Isso dito, caso haja interesse por parte do parceiro, será disponibilizado como código fonte, formato de software livre sem garantias. Licença GPLv2+. O parceiro deverá instalar suas próprias chaves criptográficas para validar *firmware*, e servidor de atualizações (não fornecido).

Todos os módulos são passíveis de alteração para facilitar a integração, incluindo remoção ou substituição de componentes.

Não é necessária a camada LuCI e JSON-RPC para funcionamento do medidor, pode ser removida por completo, mas nesse caso alguma coisa pequena precisa ser desenvolvida em substituição caso haja qualquer interface de controle/comando entre o *firmware* e o motor de medição.

Sistema operacional

O SIMETBox, no momento, funciona e está testado e validado nos seguintes sistemas operacionais:

- OpenWRT 15.05, arquitetura ar71xx
 - Mínimo 8MiB FLASH, 32MiB RAM
 - Absolutamente necessários vários *backports* de alterações provenientes de versões mais recentes do OpenWRT, que a equipe NIC.br disponibiliza no GitHub
 - Alguma funcionalidade é perdida, por *bugs* no OpenWRT 15.05: JSON-RPC para conexão com App mobile. Algumas medições suplementares.
 - ATENÇÃO: usa OpenSSL antigo, irá causar problemas na camada TLS eventualmente.
- OpenWRT 18.05, arquitetura ar71xx
 - Mínimo 8MiB FLASH, 64MiB RAM
 - Recomendado 16MiB FLASH
 - Absolutamente necessários vários *backports* de alterações provenientes de versões mais recentes do OpenWRT, que a equipe NIC.br disponibiliza no GitHub
 - ATENÇÃO: usa OpenSSL antigo, irá causar problemas na camada TLS eventualmente.
- OpenWRT 19.07, arquiteturas ar71xx, ath79, mt*, outras
 - Mínimo 8MiB FLASH, altamente recomendado 16MiB FLASH
 - Mínimo 64MiB RAM, recomendado 128MiB RAM para 802.11ac+ devido ao uso de RAM pelo *stack* IEEE 802.11 durante tráfego pesado.
 - Altamente recomendado alguns *backports*, que a equipe NIC.br disponibiliza no GitHub
- OpenWRT 21 a OpenWRT 23
 - Mínimo 16MiB FLASH
 - Mínimo 64MiB RAM, recomendado 128MiB RAM para 802.11ac+ devido ao uso de RAM pelo *stack* IEEE 802.11 durante tráfego pesado.

ATENÇÃO: quantidade de FLASH na lista acima é o tamanho do **chip** de FLASH utilizado. **Não é a quantidade de FLASH que o motor de medição precisa, essa informação está em outra área deste documento.** Da mesma forma, a quantidade de RAM informada acima é a quantidade de RAM total do sistema, não é a quantidade de RAM livre que o motor de medição precisa para funcionar.

Alterações necessárias no OpenWRT

A equipe do NIC.br mantém *backports* de alterações provenientes do próprio OpenWRT em versões mais atuais, para resolver *bugs* que estejam causando problemas para o

SIMETBox, bem como para (em alguns casos onde não há empecilho por parte do *kernel* mais antigo) suporte a novo hardware.

São (podem ser) necessários *backports* no: OpenWRT principal, *feed* de packages, e *feed* do LuCI.

Disponíveis em:

- <https://github.com/simetnicbr/openwrt-openwrt>
 - branches: *chaos_calmer*, *openwrt-18.06*, *openwrt-19.07* já possuem os *backports* aplicados.
 - branches: *topic/cc-backports* (*openwrt 15.05*), *topic/18.06-backports*, *topic/19.07-backports* são os *backports* isolados.
 - Estes branches sofrem *git rebase* de vez em quando.
- <https://github.com/simetnicbr/openwrt-packages>
 - branches: *for-15.05*, *openwrt-18.06*, *openwrt-19.07* já possuem os *backports* aplicados.
 - branches: *topic/** - *backports* e atualizações separadas.
 - Estes branches sofrem *git rebase* de vez em quando.
- <https://github.com/simetnicbr/openwrt-luci>
 - Desnecessário se não for usar a interface LuCI do SIMETBox, ou o JSON-RPC (*ubus*).
 - Não foi necessário nenhum *backport* para *openwrt-19.07* até agora
 - branches: *for-15.05*, *openwrt-18.06*, *openwrt-19.07* já possuem os *backports* aplicados.
 - branches: *topic/15.05-backports*, *topic/18.06-backports* são os *backports* isolados.
 - Estes branches sofrem *git rebase* de vez em quando.

O NIC.br também inclui funcionalidades independentes do motor de medição na SIMETBox quando necessário / desejado. Estas funcionalidades estão presentes no “empacotamento OpenWRT” (*simetbox-openwrt-feed*, descrita em outra seção deste documento), como pacotes separados. Em todos os casos, são *software livre* e disponíveis no formato de código fonte.

BUSYBOX

Muitos dos utilitários utilizados pela parte em *shell* do motor de medição, inclusive o próprio *shell*, são implementados pelo BUSYBOX do próprio OpenWRT.

O BUSYBOX do OpenWRT deve ser configurado com algumas opções diferenciadas do padrão [do OpenWRT], para adicionar algumas funcionalidades necessárias para medições **auxiliares** do SIMET. O custo em espaço de FLASH é muito menor que o que incorreríamos ao duplicar essa funcionalidade no motor de medição.

Além dessas opções de configuração do BUSYBOX, eventualmente pode ser necessário algum *patch* para adicionar funcionalidade nova ou corrigir funcionalidade antiga do BUSYBOX sem atualizar o mesmo completamente.

Como não há alterações críticas pendentes, trataremos delas no futuro. De qualquer forma, os patches implementando essa funcionalidade serão incluídos no pacote de código fonte enviado por nós, enquanto não forem enviados ao *upstream* do BUSYBOX.

CONFIG do BUSYBOX (OpenWRT 18.06+):

- CONFIG_BUSYBOX_CUSTOM=y
- CONFIG_BUSYBOX_CONFIG_TRACEROUTE=y
- CONFIG_BUSYBOX_CONFIG_FEATURE_TRACEROUTE_USE_ICMP=y
- CONFIG_BUSYBOX_CONFIG_FEATURE_TRACEROUTE_VERBOSE=y
- CONFIG_BUSYBOX_CONFIG_TRACEROUTE6=y
- CONFIG_BUSYBOX_CONFIG_SHA256SUM=y
- CONFIG_BUSYBOX_CONFIG_MD5SUM=y
- CONFIG_BUSYBOX_CONFIG_FLOCK=y
- CONFIG_BUSYBOX_CONFIG_IFPLUGD=y
- CONFIG_BUSYBOX_CONFIG_FEATURE_FIND_MTIME=y

CONFIG adicional do BUSYBOX (OpenWRT 15.05)

- CONFIG_BUSYBOX_CONFIG_FEATURE_IP_ADDRESS=y
- CONFIG_BUSYBOX_CONFIG_FEATURE_IP_LINK=y
- CONFIG_BUSYBOX_CONFIG_FEATURE_IP_ROUTE=y
- CONFIG_BUSYBOX_CONFIG_FEATURE_IP_RULE=y
- CONFIG_BUSYBOX_CONFIG_FEATURE_IP_TUNNEL=y
- CONFIG_BUSYBOX_CONFIG_IP=y

LIBCURL e CURL, TLS

O motor de medição faz uso extensivo do *cURL* (e *libcurl* no código em C) para acesso às APIs web, em *https*.

É necessário que os pacotes *curl* e *libcurl* sejam configurados no OpenWRT com suporte a **cookies**, **HTTPS**, **TLS1.2** (quando possível, **TLS1.3** também) e que certificados sejam instalados para uso, através do pacote “*ca-bundle*” ou equivalente.

A SIMETBox usa os backends de criptografia “*mbedtls*” ou “*openssl*” dependendo da quantidade de *FLASH* livre. O backend “*wolfssl*” também pode ser utilizado, mas precisa de um *backport* para suportar a CA “Let’s Encrypt”.

LIBCURL config (OpenWRT):

- CONFIG_LIBCURL_COOKIES=y
- CONFIG_LIBCURL_CRYPT_AUTH=y
- CONFIG_LIBCURL_FILE=y
- CONFIG_LIBCURL_HTTP=y

Certificate store (pacote *ca-bundle*)

Para acesso aos servidores NIC.br e servidores de medição, são necessários os seguintes certificados raiz na *certificate-store* do sistema (OpenWRT):

- Certificados raízes atuais da CA “Let’s Encrypt”
- Certificados raízes atuais da CA “Comodo”
- Certificados raízes atuais da CA “Sectigo”
- Certificados raízes atuais da CA “DigiCert”
- Certificados raízes atuais da CA “ZeroSSL”

Das lições aprendidas com a expiração em 2020-05-30 da “AddTrust External Root CA”, e em 2021-09-30 da “DST Root CA X3”:

- **Remover certificados já expirados ou que vão expirar em breve da *certificate store* do sistema, particularmente o “DST Root CA X3”, para evitar problemas com OpenSSL, GNUTLS e WolfSSL do OpenWRT 15 a 18.06.**
- **Estar preparado para atualizar a *certificate-store* (junto com o firmware) periodicamente. Essa é considerada uma atualização de segurança.**

Na prática, recomendamos, **quando é possível**, instalar a versão mais atual do pacote “*ca-bundle*” do OpenWRT (isso implica *backports* de atualizações da versão mais recente do OpenWRT), já que inclui todos os certificados.

Estamos trabalhando para mapear todos os certificados-raiz necessários, e eventualmente adicionaremos um pacote com o conjunto mínimo de certificados-raiz necessários para o SIMET2 que poderá ser utilizado como alternativa ao *ca-bundle*. Até lá, recomendamos o uso do *ca-bundle* completo.

Motor de criptografia (mbedtls, wolfssl, openssl)

O OpenWRT pode ser configurado com um ou mais “motores de criptografia” (bibliotecas criptográficas). Em geral, apenas um desses motores é utilizado, para economizar espaço em FLASH.

Em ordem de espaço ocupado em FLASH, o menor motor (e com funcionalidade mais restrita) é o **mbedtls**, seguido do intermediário **wolfssl**, e por último o maior motor (e com mais funcionalidades), o **openssl**.

No momento o motor de medição SIMET exige a presença do **openssl**. Entretanto, há relatos de que as alternativas padrão do OpenWRT 22+ (wolfssl, mbedtls) são **mais rápidas** que o **openssl**, havendo portanto possível razão para incluir mais de uma biblioteca criptográfica no *firmware* quando o espaço em FLASH for suficiente.

Em versões mais antigas do OpenWRT (até a versão 17) é recomendado utilizar o **openssl** (com as devidas atualizações de segurança, sempre que possível).

Motor de Medição SIMET2

SIMET-MA é o nome do motor de medição SIMET2 para sistemas operacionais baseados em Linux. Ele é utilizado na SIMETBox, integrado ao OpenWRT. Também é utilizado nos “medidores Linux nativos”, onde é integrado com as distribuições Debian e Ubuntu (adicionaremos outras no futuro).

Usamos o SIMET-MA com sistema operacional Debian/Ubuntu em servidores, *desktops* e embarcados “maiores” como *Raspberry Pi* e similares. Ele também serve como base para o motor de medição para Android, usado pela nossa APP Mobile.

Este documento apenas explora o SIMET-MA integrado ao OpenWRT. Os “medidores nativos Linux” estão fora do escopo deste documento.

Licenciamento

O motor de medição SIMET2 (“SIMET-MA”) é *software livre*, licenciado sob a GNU Public License version 3 (GPL-3), e é propriedade intelectual do NIC.br.

O SIMET-MA utiliza outros componentes de *software* (libcurl, libubox, libjson-c, libevent, lmapd, etc) que são licenciados com licenças de *software livre* compatíveis com a GPL-3.

O licenciamento do SIMET-MA é compatível com o licenciamento do OpenWRT e seus componentes, e não introduz nenhum risco jurídico **extra** relacionado a licenciamento de *software*, quando embarcado em um *firmware* de terceiros baseado em OpenWRT.

Isso significa que, **se o *firmware* de terceiro já resolveu adequadamente a situação de licença entre eventuais componentes “proprietários” que possua e o resto do OpenWRT**, o motor de medição SIMET2 não altera essa situação.

Para isso, recomendamos que quaisquer **componentes** do *firmware* de terceiros que sejam incompatíveis com a GPL-3, utilizem única e exclusivamente a interface **UBUS** para comunicação com o motor de medição SIMET2.

ATENÇÃO: a presença ou utilização do motor de medição SIMET2 ou *firmware* SIMETBOX em um dispositivo ou serviço não garante, nem dá direito especial de acesso ou de uso, a nenhuma parte às APIs, serviços e sistemas do NIC.br necessários para a operação do motor de medição.

Embarque do motor de medição em equipamentos

O motor de medição SIMET2 **deve obrigatoriamente ser atualizado periodicamente** em toda a frota de equipamentos. O fabricante do equipamento ou do *firmware* **deve necessariamente incluir mecanismos que garantam esta atualização, bem como a identificação pelo NIC.br do fabricante e versão do *firmware* onde o motor de medição foi embarcado, e desativação remota do motor de medição.**

Desta forma, em caso de problemas, o NIC.br poderá desativar versões muito antigas ou defeituosas do motor de medição de forma “limpa”, quer seja por característica inerente do motor de medição em si, ou situação gerada pela integração do mesmo com o *firmware* de terceiros ou outra situação externa às partes.

A frequência esperada de atualização do motor de medição embarcado em *firmware* de terceiros é de até duas atualizações por ano, e serão informadas pelo NIC.br através dos contatos técnicos entre as partes.

O NIC.br poderá publicar atualizações do motor de medição com frequência maior, a seu critério, que o fabricante pode, a seu critério, adotar (recomendamos que a equipe do NIC.br seja contactada nesse caso).

Atualizações de segurança

No caso do NIC.br identificar (ou ser informado de) problemas de segurança **no motor de medição**, irá comunicar ao fabricante, de forma tempestiva **e através dos contatos técnicos definidos entre as partes**, a existência da situação, e informações relevantes.

O NIC.br não tem controle sobre períodos de embargo definidos por partes externas possivelmente envolvidas nesse processo, por exemplo, pesquisadores de segurança que tenham identificado e comunicado ao NIC.br sobre o problema de segurança.

O NIC.br não irá embargar as atualizações do SIMETBox ou outros *firmwares* devido a um problema de segurança: cabe ao fabricante ou integrador de *firmware* responder de forma tempestiva às notificações, e atualizar a sua frota de equipamentos.

O NIC.br não tem responsabilidade sobre problemas de segurança oriundos de componentes do sistema operacional e/ou componentes do OpenWRT, ou quaisquer outros componentes do *firmware*, mesmo que tenham sido adicionados unicamente para permitir o embarque ou funcionamento do motor de medição SIMET.

Equipamentos em fim de vida útil (End Of Life)

O fabricante deverá, **obrigatoriamente**, publicar e disseminar uma última atualização do seu *firmware*, desativando (de preferência, removendo) o motor de medição SIMET2 de equipamentos que não mais sofrerão atualizações de *firmware*.

Deverá ainda informar o NIC.br do fato, com detalhes exatos de modelos envolvidos, para que o NIC.br possa, quando necessário, desativar o acesso dos motores de medição dos mesmos aos seus sistemas.

Desta forma, fica preservada a exigência do NIC.br de que equipamentos com o motor SIMET2 embarcado e ativo tenham seu motor de medição atualizados periodicamente.

Geolocalização

Para que o NIC.br possa utilizar o resultado das medições (de forma anonimizada e agregada!) para a geração de relatórios sobre a qualidade do acesso Internet, é necessário atribuir a medição a um **setor censitário** (do IBGE).

Para tanto, é necessária a **geolocalização aproximada** da medição, que o NIC.br “reduz” para o **setor censitário** ao armazenar os dados de medição.

Na SIMETBox, essa geolocalização é realizada via *scan* WiFi em *background*, de baixa prioridade, para obter os BSSID de APs próximos (com *opt-out* de ESSIDS com sufixo “_nomap” ou *hidden*), e consulta a um serviço de terceiro (no momento, Google).

O resultado da geolocalização é comunicada ao usuário via interface web da SIMETBox.

Note que o *scan* para geolocalização pode interferir com o tráfego WiFi do equipamento em caso de *bugs* nos drivers do chipset de rádio ou na pilha IEEE802.11 do kernel.

Para minimizar o consumo de API paga de geolocalização e a chance de interferência com tráfego Wifi, a geolocalização do medidor é realizada de forma amostral e infrequente, ou seja, nem sempre o medidor obterá permissão para geolocalizar via “API key” do SIMET, e o próprio medidor tenta minimizar essas requisições.

De forma geral, recomendamos que o módulo de geolocalização via WiFi da SIMETBox seja DESABILITADO em firmware de terceiros. Caso haja interesse em mantê-lo habilitado no firmware, podemos verificar os detalhes: entre em contato conosco.

A geolocalização via WiFi precisa rodar como *root*, ver “integração com o sistema operacional”, abaixo.

Integração com o Sistema Operacional

O motor de medição em si não é auto-contido, ele precisa de uma fina camada de integração com o sistema operacional que deve:

1. Disponibilizar o ambiente *shell* e utilitários utilizados pelo motor de medição
 - a. bash ou busybox ash
 - b. ambiente POSIX padrão (sed, awk, head, cut, tail, tr, wc, ...)
 - c. md5sum, sha256sum
 - d. nslookup, traceroute, traceroute6, ping, ping6
 - e. (opcionais): fping, drill ou dig
2. Criar diretórios e ajustar permissões em áreas voláteis no *boot*
3. Executar o motor de medição no *boot*
 - a. reiniciar o motor de medição em caso de problemas (muito raro)
 - b. parar o motor de medição caso o usuário desabilite o mesmo
 - c. parar o motor de medição durante atualizações do sistema
4. Executar alguns componentes do motor de medição como *root* (ex: via cron)
 - a. (opcional) geolocalização via WiFi
 - b. (opcional) *self-healing* de permissões de acesso a arquivos
 - c. (opcional) reiniciar motor de medição caso tenha parado
5. (opcional) permitir ao usuário do equipamento interagir com o motor de medição

Em geral, a maior parte do esforço de integração está no último item. Sem ela, toda interação com o motor de medição terá que ocorrer através dos portais do NIC.br.

Cabe lembrar que a SIMETBox já possui toda essa integração disponível para OpenWRT + LuCI, em português, que pode ser utilizada ou servir como base de referência.

Privilégios *root* e o motor de medição SIMET2

O motor de medição tem alguns poucos componentes (escritos em *shell*, facilmente auditáveis) que precisam rodar como *root*:

- *geolocalização* (funcionalidade que pode ser desativada e removida)
- correção de permissões na filesystem (self-heal) que provou ser útil no passado para corrigir erros operacionais
- *scripts* de instalação/upgrade do openwrt
- *scripts* de inicialização que executam o medidor.

A maior parte do motor de medição roda como um usuário sem maiores permissões, “nicbr-simet”. Em particular, toda a parte “controlada remotamente” do motor de medição roda dentro desse usuário sem privilégios especiais, e portanto não tem acesso *root*.

O NIC.br não tem a possibilidade de rodar comandos fornecidos remotamente como usuário *root*. O motor de medição não faz download de *scripts*, apenas de um *schedule* declarativo que permite rodar apenas alguns poucos comandos pré-autorizados e previamente embarcados no *firmware*, sempre como usuário “nicbr-simet”.

Para que isso seja possível, os utilitários que exigem *root* para funcionar (*traceroute*, *ping*, *fping*) são instalados com privilégio *suid root* no OpenWRT. Note que são utilitários / aplicativos do sistema operacional, e não fornecidos pelo NIC.br. Em todos os casos, estes utilitários fazem uso de boas práticas quando *suid root* (eles foram projetados para poderem ser utilizados desta forma) e desativam o privilégio mais elevado assim que possível.

Note que foram necessárias algumas correções no *fping* (removia privilégio *root* cedo demais para a opção `SO_BINDTODEVICE` do Linux, necessária em IPv6 link-local), e no empacotamento do `BUSYBOX` pelo OpenWRT. Essas correções foram aceitas por ambos os projetos e incluídas nas versões mais recentes dos mesmos, e como são críticas para o funcionamento do motor de medição, estão incluídas nos *backports* do OpenWRT mantidos pelo NIC.br que já mencionamos anteriormente.

Pacotes OpenWRT do motor de medição

A SIMETBox padrão integra o motor de medição SIMET-MA ao OpenWRT utilizando uma *feed* de pacotes separada (terminologia do `openwrt`. *feed* é um repositório de pacotes), chamada *simetbox-openwrt-feed*, e disponível no GitHub.

Esta *feed* contém vários pacotes relevantes para o SIMETBox. **Nem todos são relevantes para a integração em firmware de terceiro.**

Os pacotes relevantes para a integração são pequenos, a tabela seguinte mostra, **a título de exemplo**, os tamanhos dos pacotes IPK em um build para OpenWRT 19.07, MIPS 24kc, otimização padrão do OpenWRT.

Nome	Tamanho (bytes)	Descrição
<code>simetbox-openwrt-simet-lmapd_0.15.1-1_mips_24kc.ipk</code>	43242	motor de medição: scheduler LMAP
<code>simetbox-openwrt-simet-ma_0.17.6.3-1_mips_24kc.ipk</code>	76581	motor de medição: componentes principais e integração deles com o <code>openwrt</code> .
<code>simetbox-openwrt-base_108-1_mips_24kc.ipk</code>	5870	motor de medição: parte da integração com o <code>openwrt</code> . Ainda possui algum branding SIMETBox
<code>simetbox-openwrt-luci_1.7-1_mips_24kc.ipk</code>	10925	Opcional Extensão da interface web do OpenWRT (LuCI) para controle do motor de medição e informações sobre a SIMETBox, a ser usada como referência
<code>firewall-urpf_1-1_all.ipk</code>	3249	Opcional Firewall anti-spoofing de origem, e módulo

		LuCI para configurar.
--	--	-----------------------

Os pacotes relevantes são:

- `simetbox-openwrt-simet-lmapd`
 - Requer `libevent2`, `libjson-c`
 - Implementa o *scheduler* de medições
- `simetbox-openwrt-simet-ma`
 - Requer `libcurl`, `curl`
 - Requer `ca-bundle` ou subconjunto de certificados necessários para acessar as APIs do NIC.br.
- `simetbox-openwrt-base`
 - Iremos eliminar / reduzir para facilitar a compreensão do código fonte: muito do código nela é do SIMET1 (medidor legado), e estará inativo (não é instalado) na configuração do motor de medição que propomos nesse documento.
 - Contém boa parte da configuração inicial SIMETBox, a ser alterada ou removida no firmware de terceiro.
 - Requer: `libjson-c`, `luci-lib-jsonc`, `luci-lib-json`, `libuci-lua`
- (opcional) `simetbox-openwrt-luci`
 - Está com branding SIMETBox, serve apenas como exemplo
 - Requer algumas bibliotecas lua presentes no `simetbox-openwrt-base`

Outros pacotes do Sistema Operacional

O motor de medição SIMET2 faz uso de diversos utilitários do sistema operacional, mas quase todos podem providos via BUSYBOX e já estarão disponíveis em uma instalação típica do OpenWRT.

Algumas medições auxiliares do SIMET2 fazem uso de utilitários específicos:

- “`dig`” ou “`drill`”, para medições avançadas do DNS e DNSSEC (“`drill`” é menor), opcional porém desejado
- “`nslookup`”, “`traceroute`”, “`ping`”, e suas versões IPv6 (pode ser do `busybox`)
- “`fping`”

Além das medições, o motor de medição em si precisa de alguns utilitários para que sua parte em *shell* possa funcionar:

- “`curl`” com suporte `https` e certificados
- “`sed`”, “`awk`”, “`tr`”, “`md5sum`”, “`sha256sum`”, “`grep`”, “`wc`”, “`df`”, ... (`busybox`)

Código fonte

O código fonte do motor de medição, e do empacotamento OpenWRT (“SIMETBox”) são mantidos de forma separada.

Preparação de ambiente, e recomendações

1. Inicialmente ao se acostumar com o SIMETBox, use o OpenWRT, inclusive as feeds “packages” e “luci”, disponibilizadas pela equipe SIMET no GitHub. Evita problemas, já que existem *backports* que são críticos para o bom funcionamento do mesmo.
2. Utilize a *simetbox-openwrt-feed*, pelo menos inicialmente, para trabalhar com o motor de medição SIMET integrado ao OpenWRT.
3. Para esse documento, assumimos que o código fonte está em `/usr/src/simetbox/` ao dar exemplos de arquivos, symlinks e configurações.
4. Sugerimos que utilize um sistema, container ou VM “Debian GNU/Linux versão 10 ou versão 11”, atualizada, para trabalhar com o código fonte. O motor de medição em si tolera bem Debian 8 ou mais recente, Ubuntu LTS 16 ou mais recente. Mas o mesmo **não** pode ser afirmado do OpenWRT.
5. Independente de que sistema operacional seja utilizado para desenvolvimento, recomendamos utilizar containers docker ou uma VM com Debian 10 para builds do OpenWRT nesse momento: é muito mais estável, já que é o sistema operacional que o `openwrt.org` está usando no momento nos builders deles.
6. Assuma-se familiaridade com git (<https://git-scm.org>, <https://git-scm.com/book/en/v2>), e openwrt.

Ambiente de build e desenvolvimento

Prepare de antemão um ambiente que compila o *openwrt* na versão desejada e testado (gera imagens *openwrt* funcionais para o seu *hardware*). Isso facilita **tudo**. Depois, troque pelo *fork* *openwrt* (*openwrt*, *packages* e *luci*) usado no SIMETBox e veja se ainda compila usando a mesma config que você está acostumado.

Sugerimos utilizar um *container* ou VM, e compilar do código fonte completo (sem usar o SDK binário) por motivos de estabilidade.

Recomendamos que somente após essa parte básica de *openwrt* estar funcionando, ative a *feed* *simetbox-openwrt-feed*, configs, e tente compilar o SIMETBox em si.

Obtendo o código fonte

```
mkdir /usr/src/simetbox ; cd /usr/src/simetbox/
```

```
git clone https://github.com/simetnicbr/simet-ma.git
```

```
git clone https://github.com/simetnicbr/simet-lmapd.git
```

```
git clone https://github.com/simetnicbr/simetbox-openwrt-feed.git
```

```
git clone https://github.com/simetnicbr/openwrt-openwrt.git
```

```
git clone https://github.com/simetnicbr/openwrt-packages.git
```

```
git clone https://github.com/simetnicbr/openwrt-luci.git
```

Preparação do código fonte

Não existe preparação específica para o SIMETBox. Descrevemos abaixo pontos importantes em uma preparação típica para desenvolvimento local usando openwrt.

1. Lembre-se de alterar a configuração de *feeds* em **openwrt-openwrt/feeds.conf** para que utilize as *feeds* de *packages* e *luci* com *backports* da equipe SIMET, e para adicionar a *simetbox-openwrt-feed* bem como quaisquer outras feeds que vocês precisem.
2. Lembre-se que OpenWRT gosta de *paths* absolutos (a partir da raiz da filesystem) em URLs e *symlinks*. Por exemplo, “src-cpy ../openwrt-packages/” no `openwrt-openwrt/feeds.conf` parece funcionar em openwrt-19.07, mas é mais seguro utilizar o caminho absoluto. **Para esse documento, assumimos que o código fonte está em em /usr/src/simetbox/ ao dar exemplos de arquivos, symlinks e configurações.**
3. **OPCIONAL:** Crie os *symlinks* `git-src` (sempre usando caminho absoluto) se quiser usar o código fonte local no lugar de usar uma tarball criada de um git-clone limpo feito pelo build-system do openwrt. Nesse modo, no lugar do *openwrt* criar uma tarball do código fonte (etapa de “download” do build), ele faz um “git checkout” do *branch ativo* do repositório `.git` apontado pelo *symlink*.
 - a. Nesse caso lembre-se de ativar a opção correta no `.config` do openwrt, ou os *symlinks* `git-src` não serão respeitados!
 - b. Lembre que é usado um *novo checkout* do conteúdo do repositório apontado pelo *symlink*, alterações que não estão ainda em um *commit* não são usadas.
 - c. E lembre-se de ter o branch desejado em checkout naquele repositório...
4. Instale a *feed packages*, *luci*, e *simetbox-openwrt-feed*, e ative os pacotes delas, procedimento do openwrt usando “./script/feeds”.

exemplo de openwrt-openwrt/feeds.conf :

```
src-cpy packages /usr/src/simetbox/openwrt-packages/  
src-cpy luci /usr/src/simetbox/openwrt-luci/  
src-cpy simetbox /usr/src/simetbox/simetbox-openwrt-feed/  
#src-git simetbox https://github.com/simetnicbr/simetbox-openwrt-feed.git  
#src-git packages https://github.com/simetnicbr/openwrt-packages.git;openwrt-19.07  
#src-git luci https://github.com/simetnicbr/openwrt-luci.git;openwrt-19.07
```

Motor de medição:

Recomendação: leia bem por cima a seção “motor de medição” só para fazer uma ideia geral, e vá para a seção “empacotamento do motor de medição no openwrt”, começar de lá é mais simples.

O motor de medição é composto de dois módulos:

- simet-lmapd, mantido separado do resto do código do motor de medição porque é baseado na implementação de referência “lmapd”, da IETF.
 - <https://github.com/simetnicbr/simet-lmapd>
- simet-ma, contém o motor de medição exceto “lmapd”.
 - <https://github.com/simetnicbr/simet-ma>
 - Contém o empacotamento Debian/Ubuntu também, e o empacotamento *docker*. Toda a parte de integração com o sistema operacional (*inclusive do simet-lmapd*) Debian/Ubuntu, e suporte *docker*, está no “simet-ma”.
 - Contém (no diretório “cmake-projects”) alguns componentes nativos do OpenWRT para uso em outros sistemas operacionais como Debian, Ubuntu, etc. **Em um build “simetbox” (leia-se: para openwrt), essa parte do código fonte é ignorada:** usamos os componentes fornecidos pelo próprio openwrt.

O motor de medição é escrito em C e *shell* (bash), e funciona em Linux, 32-bits e 64-bits, *little-endian* e *big-endian*.

O código fonte utiliza CMAKE e AUTOMAKE dependendo do submódulo.

O simet-ma tem um *hack* horrível no diretório “cmake-projects” para compatibilizar automake com cmake. **Não perca tempo tentando entender esse *hack*. Essa parte do código fonte é apenas uma cópia de parte da “libubox”, e do utilitário “jsonpath” do openwrt, e irrelevante em *builds* para openwrt.**

Durante um build openwrt, o simet-ma *não* irá utilizar o diretório “cmake-projects”, ele usa a *libubox* no diretório de includes do openwrt, e depende do *jsonpath* compilado pelo próprio openwrt.

Recomendamos que os engenheiros de firmware que não conheçam bem o CMAKE e AUTOMAKE, olhem como o build é feito nos “Makefile” do OpenWRT (ver na parte de empacotamento openwrt, abaixo).

É necessário configurar corretamente o build de cada componente para que seja adequado à finalidade (Debian/Ubuntu, *docker*, ou OpenWRT). Partes do código fonte são selecionadas para build e instalação de acordo com o tipo do build (se é “simetbox”, ou seja, para openwrt, ou se não é “simetbox”, ou seja, é para alguma outra coisa que não o openwrt).

Recomendamos **veementemente** que olhe como o build é disparado nos Makefiles da *feed* openwrt. Em particular, um build openwrt do simet-ma deve utilizar “--enable-simetbox” na chamada do “configure”, e o s builds do simet-lmapd precisam de

opções de compilação para ajustar a *path* padrão dos *schedules*, sem o qual ele não funcionará em conjunto com versão recente do *simet-ma*.

O motor de medição é desenvolvido em ambiente **Linux**, Debian 12 e Ubuntu LTS 22. Versões mais antigas são suportadas de forma limitada.

simet-ma

- diretórios *debian/*, *docker/*
 - empacotamento *debian/ubuntu* e *docker*. O suporte *docker* é preliminar e pode não estar plenamente funcional.
- diretório *cmake-projects/*
 - cópia de trecho do *libubox* do *openwrt* para build *debian/ubuntu/docker*
 - cópia de *jsonpath* do *openwrt* para build *debian/ubuntu/docker*
 - não utilizado em builds para *openwrt*
- diretório *simet-runner/*
 - componente em *shell* que roda a medição principal (instalado como “*simet-ma_run.sh*”, rodá-lo no medidor realiza uma medição base)
 - executado diretamente ou via “*lmapd*”
 - eventualmente será removido, mas faltam ainda algumas funcionalidades no “*lmapd*” para que isso seja possível
- diretório *twamp-client-c/*
 - medição TWAMP (two-way delay, jitter, ordering, packet-loss)
 - executado pelo “*simet-runner*” ou diretamente pelo “*lmapd*”
- diretório *tcpbw-client-c/*
 - medição de vazão TCP
 - executado pelo “*simet-runner*”
- diretório *inetup-client-c/*
 - medição de disponibilidade de conexão internet
 - medição de quantidade de bytes trafegados na wan (opcional)
 - deve ser executado e reiniciado pela integração com o sistema operacional (ex: por */etc/init.d/simet-ma* nos empacotamentos *openwrt* e *debian*)
- diretório *simet-api/*
 - componentes em *shell* do motor de medição, e API de integração (comunicação do sistema operacional e UIs com o motor de medição).
 - *lmap-tasks/*, *config.d/*, *registry.d/*
 - scripts de medições auxiliares executadas via “*lmapd*”, configuração de linha de comando de componentes do motor de medição que o “*lmapd*” pode executar, e lista de executáveis que o “*lmapd*” aceitará executar.
 - se não estiver mencionado em *registry.d/*, o “*lmapd*” **não** aceita executar um script ou programa. Componentes não instalados podem ser mencionados por entradas em *config.d/* e *registry.d/*: o “*lmapd*” irá ignorá-los ao notar que não existe o executável no sistema.
 - *conf/*

- configuração padrão do motor de medição e *schedule* padrão do motor de medição, depende do tipo de *build* (simetbox/openwrt, ou outro).

Em geral, arquivos com o nome prefixado por “simet_” são usados tanto para simetbox/openwrt quanto para outros builds, *exceto* quando existe o mesmo arquivo com prefixo “simetbox”. **A distinção do que usar no build dependendo do tipo (simetbox/openwrt ou não-openwrt), ou do que deve ser instalado pelo build dependendo do tipo é comandada pelo “Makefile.am”, não é baseada em heurística de nome de arquivo.**

Idéia geral de como a integração do sistema operacional com o simet-ma funciona:

O arquivo específico para o sistema operacional (por exemplo, /etc/init.d/simet-ma, /etc/init.d/simet-lmapd, /etc/cron/simet-ma, ou uma *unit systemd*) **fornecido pelo empacotamento para aquele sistema operacional**, tenta ao máximo utilizar apenas algum dos scripts “simet_*” ou “simetbox_*” presentes no diretório simet-api/ do simet-ma.

Estes scripts presentes no diretório simet-api/ são instalados em /usr/bin (openwrt) ou /opt/simet/bin (Debian, Ubuntu) pelo empacotamento (descrito na próxima seção).

Existem, portanto, scripts de “api” do motor de medição, que são uma interface mais estável, para que outros scripts/utilitários diretamente ligados ao *firmware* (fornecidos no empacotamento) executarem.

Infelizmente essa abstração não é completa. Nesse caso, o melhor é o script do empacotamento utilizar o “simet_lib_config.sh” (presente em simet-api/, instalado em /usr/lib/simet/ ou /opt/simet/lib/simet/) para obter as diversas definições de diretórios e nomes de arquivo que o motor de medição irá utilizar.

No empacotamento openwrt, há uma camada extra (implementação *rpcd simet-ma*, acessível via *ubus*). Essa camada extra precisa de muito conhecimento de detalhes internos do motor de medição e acaba sendo atualizada junto com o motor de medição, então ali usamos nomes de arquivo e interpretamos o conteúdo deles diretamente. Como faz parte do empacotamento, ela sabe exatamente onde o empacotamento configurou o motor de medição para colocar arquivos.

Empacotamento e build openwrt do motor de medição

O empacotamento para openwrt fica inteiramente contido na *simet-openwrt-feed*, disponível em:

<https://github.com/simetnicbr/simetbox-openwrt-feed>

Esse documento assume que há conhecimento prévio de como funciona um *build* do OpenWRT, e as *package feeds*.

Cada diretório da *feed* é um pacote separado. Como essa *feed* é para o SIMETBox inteiro, não só o motor de medição, ela ainda contém alguns pacotes que vocês podem ignorar.

Pacotes a ignorar

- “easycwmp”, “libmicroxml”
 - agente TR-069 em software livre, opcional
- “firewall-urpf”
 - extensão opcional do LuCI para melhorar a firewall padrão do OpenWRT com proteção anti-spoofing de origem (IETF BCP-38).
 - não faz parte do motor de medição, mas se acharem interessante, sintam-se livres para utilizar
 - a proteção uRPF ativada por essa extensão da firewall custa um *route lookup* extra para cada pacote sainte na WAN: não sai de graça em performance.
- “simetbox-openwrt-availability-config”
 - componente legado do SIMET1, a ser removido
- “simetbox-openwrt-config”
 - componente legado do SIMET1, a ser removido
- “simetbox-openwrt-ntpd”
 - configuração legada de NTP a ser revisada
- “simetbox-openwrt-zabbix”
 - configuração legada de Zabbix, NIC.br apenas

Tour do “empacotamento” para openwrt do motor de medição

Comece olhando o pacote **simetbox-openwrt-simet-lmapd**, que é o empacotamento mais simples.

simetbox-openwrt-simet-lmapd

<https://github.com/simetnicbr/simetbox-openwrt-feed/tree/master/simetbox-openwrt-simet-lmapd>

- Leia o README, ele mostra onde partes importantes do “lmapd” estarão localizadas usando esse pacote (ajuda a entender o Makefile e o *script* em files/etc/init.d/simet-lmapd)
- Leia o Makefile. Note os parâmetros que são passados para o **cmake** para configurar o build. Note a parte de install, que cria diretórios e move arquivos para seus locais finais.
- Em files/ você encontra os *scripts* de integração que serão instalados no firmware.

/etc/init.d/simet-lmapd

Registra um “serviço procd” (terminologia OpenWRT) que executa o “lmapd” no boot, e permite reiniciar, parar, iniciar, e fazer reload de configuração.

- O pacote contém também dois *patches* para compatibilidade com o CMAKE absurdamente antigo do OpenWRT 15.05, o Makefile detecta que serão necessários e os aplica.

Como o README do simetbox-openwrt-simet-lmapd informa, a maior parte da “cola” integração do motor de medição com o OpenWRT, inclusive muita coisa referente ao *lmapd*, está dentro do pacote simetbox-openwrt-simet-ma.

O “lmapd” está separado do simet-ma em um pacote openwrt porque seu código fonte é separado do código fonte do simet-ma, apenas.

simetbox-openwrt-simet-ma

<https://github.com/simetnicbr/simetbox-openwrt-feed/tree/master/simetbox-openwrt-simet-ma>

Esse é o empacotamento principal do motor de medição. Ele contém não só a parte de interface para executar o motor de medição, mas também inclui uma camada de abstração extra, exclusiva para o OpenWRT, para interagir com o motor de medição via *ubus* (componente do OpenWRT).

O ideal seria que a comunicação do firmware com o motor de medição seja restrita às: chamadas *ubus*, e *start/stop/reload* dos serviços “simet-ma” e “simet-lmapd” via *procd* (*/etc/init.d/simet-ma*, */etc/init.d/simet-lmapd*). Essa é inclusive a direção do desenvolvimento futuro da SIMETBox, e facilitará tremendamente a atualização do motor de medição sem interferir com o resto do firmware.

- Makefile
 - O Makefile é bem simples, a única opção necessária para o *build* do *simet-ma* é “--enable-simetbox”, atente para a parte de *install*, ela dá a dica de onde o motor de medição ficará no firmware final: */usr/bin*, */etc/simet*, */usr/lib/simet*
- Config.in
 - Símbolos de configuração para o build. Recomendamos que o *firmware* de terceiros utilize o “default”, que é: medição “SIMET_BCP38” **desativada**, suporte a APPLINK SIMET **desativado** (ver próximo ítem).
- diretórios *applink-files/*, *applink-htdocs/*
 - interface de controle remoto para uso do SIMET Mobile App (“APPLINK SIMET”). Podemos conversar sobre ela depois se houver interesse.
 - Permite alterar toda a configuração do *openwrt* via remote “*ubus uci*”, bem como controlar o motor de medição já que dá acesso à interface “*ubus rpc*” descrita abaixo.
 - recomendamos deixar isso desativado em firmware de terceiros.
- diretório *files/*
 - **scripts** de integração e a interface ***ubus rpcd***
 - O NIC.br apoiará com as alterações e desenvolvimento que forem necessários no empacotamento e camada de abstração / comunicação com o motor de medição que vocês identificarem como necessários para integrar ao seu firmware.
 - *etc/init.d/simet-ma*
 - Script para *start/stop/reload/restart* do motor de medição. o *procd* do OpenWRT é utilizado como supervisor de processos, e cuida de reiniciar o motor de medição se o mesmo falhar (com proteção contra *loop de restart*).
 - lembrar que */etc/init.d/simet-lmapd* está em *simetbox-openwrt-simet-ma*.
 - *etc/config/simet-ma*

- Configuração padrão das medições, pode ser alterado via “uci” (do OpenWRT). Necessário enviar sinal de reload para o motor de medição para aplicá-las imediatamente.
 - Serão ignoradas caso os servidores centrais do NIC.br enviem configuração diferente para o medidor.
- etc/uci-defaults/99-simet-ma-crontabs
 - opcional
 - instala um *cronjob* que roda *simetbox_refersh_cron* bem como o componente *root* da geolocalização periodicamente
 - chamada à geolocalização pode ser removida se não for utilizada
 - chamada à *simetbox_refresh_cron.sh* pode ser removida se não for considerada útil
- usr/bin/simetbox_refresh_cron.sh
 - opcional, ver crontabs acima, mas é desejável.
 - garante que o medidor vai tentar registrar-se a cada 24h. Importante, porque caso algum problema ocorra no registro a ponto de parar o motor de medição, precisaria de um *reboot* para que ele tente registrar-se novamente seom esse script
 - força recarga do *schedule* caso algum defeito em um *schedule* enviado pelos servidores centrais omita a recarga periódica
- etc/hotplug.d/iface/99-simet-ma
 - Executa */etc/init.d/simet-ma reload* quando há mudança nas interfaces WAN
- usr/libexec/rpcd/simetbox, usr/bin/check_dfl_passwd.lua
 - módulo *ubus rpcd simetbox*
 - usado pelo APPLINK SIMET, será movido para dentro de *applink-files/* na próxima versão.
- usr/libexec/rpcd/simet_ma, usr/share/rpcd/acl.d/simet-ma.json, usr/share/acl.d/simet-ma.json
 - módulo *ubus rpcd simet-ma*
 - permite controle e interface com o Motor de Medição
 - atualizado pela equipe NIC.br junto com o motor de medição

OpenWRT config

1. Ver seção “Busybox” para a config desejada para o Busybox.
2. Da *simetbox-openwrt-feed*, ative apenas o SIMET2:
 - a. Network -> SIMETBox ->
 - i. *simetbox-openwrt-base* **ligado (y)**
 - ii. *simetbox-openwrt-simet-ma* **ligado (y)**
 - iii. *simetbox-openwrt-simet-lmapd* **ligado (y)**
 - b. Network -> SIMETbox -> *simetbox-openwrt-base* (Configurações) -> SIMET1 **desligado**
3. Para o *simet-ma* e o *simet-lmapd*, deixe as opções dos mesmos no “default”.
 - a. No momento, é necessário ligar o *simetbox-openwrt-base* porque o mesmo é uma dependência do *simet-ma*. Será melhorado com a remoção completa do SIMET1 em breve
4. A remoção do SIMET1 em breve irá remover também os “warnings” sobre configuração circular, etc. na *simetbox-openwrt-feed*.

Estado dos símbolos de configuração (openwrt-19.07):

A maior parte da configuração necessária para o motor de medição SIMET2, inclusive do BusyBox, é padrão no openwrt-19.07. **Esse não é o caso de versões anteriores, e pode não ser verdade para versões futuras. A seção BUSYBOX deste documento contém a configuração mais exata.**

Segue uma configuração mínima para o openwrt-19.07, a ser completada com “make defconfig”, e depois alterada com as configurações que forem desejadas para o hardware alvo.

```
CONFIG_FEED_simetbox=y
# CONFIG_PACKAGE_simet-nping is not set
CONFIG_PACKAGE_simetbox-openwrt-base=y
# CONFIG_SIMETBOX_COMMON is not set
# CONFIG_PACKAGE_simetbox-openwrt-config is not set
# CONFIG_PACKAGE_simetbox-openwrt-luci is not set
# CONFIG_PACKAGE_simetbox-openwrt-ntpd is not set
CONFIG_PACKAGE_simetbox-openwrt-simet-lmapd=y
CONFIG_PACKAGE_simetbox-openwrt-simet-ma=y
# CONFIG_SIMET2_SPOOFER is not set
# CONFIG_SIMETBOX_APPLINK is not set
# CONFIG_PACKAGE_simetbox-openwrt-zabbix is not set

CONFIG_BUSYBOX_CUSTOM=y
CONFIG_BUSYBOX_CONFIG_FEATURE_TRACERROUTE_USE_ICMP=y
```