NOC Service Activation. Документальное описание для версии 0.9. Черновик

Примечание

В этой статье я попробую подробно расписать как работает один из важнейших модулей NOC. Описание будет основываться на состоянии разработки в текущий момент. Так как произошло достаточно много изменений с момента релиза версии 0.8, я решил назвать это описанием к версии 0.9, в надежде что до ближайшего релиза больше ничего меняться не будет, либо изменения будут минимальны. Так же по ходу статьи буду делать отметки с замечаниями моментов которые требеют дополнительного пояснения от разработчиков. Итак, попробуем...

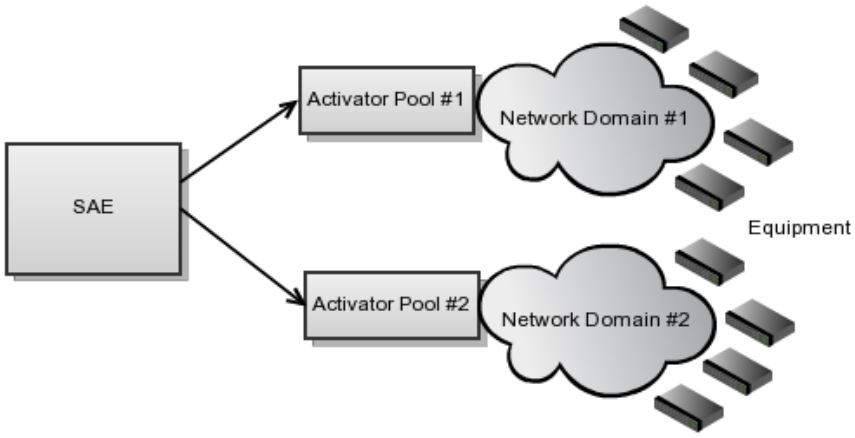
Введение.

Service Activation это один из самых важных модулей NOC, если не самый важный. Функционал данного модуля превращает NOC из обычной системы учета с набором таблиц, в мощный инструмент для автоматизации управления сетью. Service Activation обеспечивает сбор технической информации о сети и абстрагирует ее для других модулей от типа и производителя оборудования.

Архитектура.

Принципиально данный модуль разделен на две части:

1. Service Activation Engine (сокр. SAE) - центральный процесс, единая точка входа для всех задач, которые должны быть выполнены на оборудовании, распределяет все поступающие задачи между доступными активаторами
2. Activator - процесс, который непосредственно выполняет подключение к оборудованию.



Активаторы объединены в пулы для распределения нагрузки по множеству процессов, но каждое сетевое устройство может обслуживаться только одним конкретным пулом

Принцип действия.

Работает все это следующим образом. На центральном сервере запускается процесс noc-sae и начинает ждать подключения от удаленных активаторов. Процесс noc-activator запускается в месте наиболее оптимальном для управления оборудованием, подключается к SAE и проходит процедуру регистрации и аутентификации, если все настройки верные в логе активатора можно будет увидеть следующее:

noc-activator.log

```
2014-11-24 10:28:40,755 [root] Negotiation protocol 'NOC SAE PROTOCOL (http://nocproject.org/)' version '1.0'
2014-11-24 10:28:40,763 [root] Protocol version negotiated
2014-11-24 10:28:40,767 [root] Registering as 'noc'
2014-11-24 10:28:40,778 [root] Registration accepted
2014-11-24 10:28:40,779 [root] Authenticating as noc
2014-11-24 10:28:40,812 [root] Authenticated
```

Все, между активатором и SAE установлен канал связи, по которому будут отправляться задачи и возвращаться результат. Данные, передаваемые по каналу сжимаются и, что важнее, шифруются, так как там же передаются логины и пароли на доступ к оборудованию.

Белое пятно

Мне неизвестно какой алгоритм и какая длина ключа используется для шифрования, поэтому невозможно оценить степень безопасности выбранного разработчиками решения, так что я бы рекомендовал не подключать активаторы через публичные сети, а использовать более известные и проверенные решения для организации безопасного канала связи активатора с SAE

Обновление. Получен комментарий от разработчика:

```
KEY_EXCHANGES = ["diffie-hellman-group1-sha1"]
PUBLIC_KEYS = ["ssh-dss"]
CIPHERS = ["aes256-cbc", "blowfish-cbc", "3des-cbc"]
MACS = ["hmac-sha1", "hmac-md5"]
COMPRESSIONS = ["zlib"]
CIPHER_MAP = {
              "aes256-cbc": ("AES", 32),
              "blowfish-cbc": ("Blowfish", 16),
              "3des-cbc": ("DES3", 24),
}
MAC_MAP = {
           "hmac-sha1": hashlib.sha1,
           "hmac-md5": hashlib.md5
}
```

там протокол, эквивалентный ssh

по умолчанию будет AES 256 bit/HMAC SHA1

в следующих версиях будет обычный TLS

и JSON-RPC поверх HTTPS

и не будет protocol buffers

Так как инициатива по созданию подключения исходит от процесса активатора, это позволяет устанавливать связь через NAT или statefull firewall, что, в общем случае, упрощает развертывание удаленных активаторов.

Настройка.

Процесс noc-sae.

Настройка SAE процесса достаточно тривиальна, вот пример конфигурационного файла с описанием

```
[main]
##        ,    ,
logfile  = /var/log/noc/noc-sae.log
loglevel = info
logsize  = 0
logfiles = 0
syslog_host =
pidfile  = /var/run/noc/noc-sae.pid
mrt_log = false

[debug]
##     ,  ,
enable_manhole = false
enable_timing = false
timing_base = local/timing

[sae]
##   SAE
shards     = default           #   SAE    shards,          NOC sharding,     SAE     default shard
listen     = 127.0.0.1         # ip    SAE     , -   loopback , .. noc-activator          noc-sae,  SAE
0.0.0.0
port       = 19701 # TCP
refresh_event_filter = 600     #   ,
force_plaintext = 127.0.0.1/32 #            ip
max_mrt_rate_per_sae = 0       #        ,      TACACS          /
max_mrt_rate_per_shard = 0     #

[event]
strip_syslog_facility = true   #   ,
strip_syslog_severity = true   #   ,
```

Создание пулов активаторов

Чтобы завести пул активаторов, надо через Web интерфейс пройти в Service Activation > Setup > Activators, нажать Add

Name - название пула активаторов, это же имя должно быть указано в настройках каждого процесса noc-activator, который должен подключиться и получать задачи для этого пула

Shard - указывает принадлежность пула активаторов к конкретному shard, должен совпадать с тем что указан в настройках SAE, в общем случае будет всегда default как и у SAE

Prefix Table - позволяет установить ограничение из каких сетей разрешено подключение активаторов к этому пулу. По-умолчанию существует только одна таблица "Activator::default" в которой разрешены подключения только от 127.0.0.1/32, добавить собственные можно в Main > Setup > Prefix Tables

Auth String - пароль который указывается в настройках процесса активатора в поле "secret"

Is Active - должно быть отмечено галочкой у всех активных пулов. необходимо пояснение от разработчика о поведении системы при снятии этой отметки

Min. Memebers / Min. Sessions - минимальное количество подключившихся активаторов к пулу и доступному на них сессий чтобы считать пул активным (не уверен точно, нужно пояснение разработчика)

Tags - просто теги, на функционал не влияют

Процесс noc-activator.

Попробуем разобраться с настройками активатора, тут все много интересней чем SAE.

Для начала небольшое пояснение, все процессы NOC которые работают на конкретном сервере запускаются с помощью noc-launcher, его можно строить так чтобы он запускал более одного процесса конкретного демона, это необходимо для распараллеливания нагрузки на многоядерных серверах:

noc-launcher.conf

```
[noc-activator]
enabled = true
user = root
group =
config.0 = etc/noc-activator.conf
config.1 = etc/noc-activator.conf
config.2 = etc/noc-activator.conf
config.3 = etc/noc-activator.conf
config.4 = etc/noc-activator.conf
config.5 = etc/noc-activator.conf
```

Число означает номер инстанса, то есть копии процесса, на конкретном сервере, эти номера используются при настройке активатора.

noc-activator.conf

```
[main]
#
logfile = /var/noc/log/noc-activator.{{instance}}.log
loglevel = info
logsize = 5000000
logfiles = 9
syslog_host =
pidfile = /var/noc/log/run/noc-activator.{{instance}}.pid
#  ,    ,
log_cli_sessions = false
log_cli_sessions_path = /var/log/noc/cli-sessions/{{ip}}-{{script}}-{{ts}}.log
log_cli_sessions_ip_re =
log_cli_sessions_script_re =
log_snmp_traps = false

[activator]
name = noc                    #
secret = PASSWORD             #     ,      Auth string

##
ping_instance = 1             #   ,     icmp
dedicated_ping = true         #         sae,     .    , listen_instatnce        ping_instatnce
ping_count = 3                #       ,
ping_timeout = 2              #

##    Syslog  SNMP trap
listen_instance = 0           #  ,   snmp  syslog
dedicated_collector = true    #         sae,     syslog  snmp trap.    , listen_instatnce        ping_instatnce
listen_traps = 0.0.0.0        #        snmp traps  (0.0.0.0 -   ,    ip  )
listen_syslog = 0.0.0.0       #        syslog  (0.0.0.0 -   ,    ip  )

listen_pm_data = 127.0.0.1

software_update = true
max_scripts = 10
pm_data_secret =
enable_internal_trap_parser = true
[sae]
host = 127.0.0.1              # ip   SAE    , 127.0.0.1 -         SAE,  ip ,
port = 19701                  #    SAE
local_address =

[servers]
listen_http =
listen_ftp =
listen_tftp =
[ssh]
key = etc/ssh/id_rsa         #    ,       ssh
```

Ух, для начала хватит. Надо будет добавить примеров по настройке для разных случаев и описать нюансы в работе некоторых приложений в Web интерфейсе

Продолжение следует...