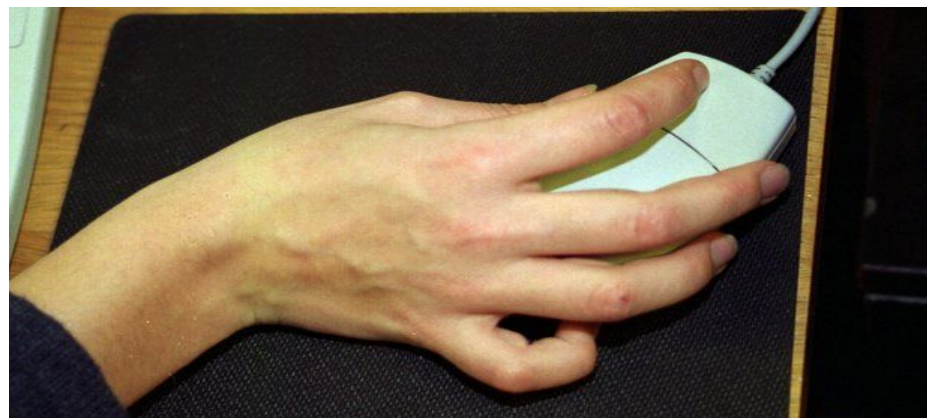


Automatisering i større sammenhæng

PROSA Stud træf
Henrik Kramshøj

Huskeregler for programmer

- Dit program er ikke alene i verden
- Det indgår i et IT-miljø med andre
- Alle har travlt
- Ingen gider lave dobbeltarbejde
- Ingen gider lave trivielt arbejde
- Automatisering er dejligt



Den gode web applikation

- Din web applikation skal være nem at gennemskue
- Gode URL'er
- og har API!
- eventuelt et API
- Det vil være godt at inkludere et API
- måske et CLI Command Line Interface med dit API
- Et API kan være gratis hvis dit framework understøtter :-D

Case: PDU'er og network equipment port descriptions

- Netværksudstyr har porte
- Netværksudstyr bruger strøm
- En port er et stik
- C13/C14 strøm Power Distribution Unit PDU
- Fiber eller RJ45 Ethernet data porte
- Typisk udstyr har to strømforsyninger, Ethernet management, konsolport serial og så et antal porte, switch/router porte - 48 porte på 1U er normalt



Hvor taster du port data?

Vi taster hardware og port data ind:

- Hardware enhedens konfiguration
set interface xe-0/0/0.123 description “Uplink VLAN123”
- IP Address Mgmt - Nipap <https://spritelink.github.io/NIPAP/>
- Network Inventory dokumentationsystem, Open Source
<https://portal.nordu.net/display/NI/NORDUnet+Network+Inventory>
- Nagios overvågning
- Så mange steder at man bliver helt træt

Enter auto-discovery og SNMP

- Nogle værktøjer HAR fundet ud af det er smart med automatisering
- Vi bruger LibreNMS, fordi det er nemt og hurtigt - henter data med SNMP og smider grafer sammen
<https://www.librenms.org/>
- Det kræver blot at SNMP kan hente data
https://en.wikipedia.org/wiki/Simple_Network_Management_Protocol
- Det er en ældre protokol SNMPv1 fra 1988 :-)

NIPAP er simpelt Web GUI

The screenshot shows the NIPAP web interface in a browser. The address bar shows the URL `127.0.0.1:5000/prefix/list#query_string=test&search_opt_parent=undefined&search_opt_child=undefined&explicit=false`. The page has a dark header with 'VRFs prefixes pools' and a 'Log out' button. A search bar contains 'test' and a blue search button. Below the search bar, it says 'Query took 0.64 seconds.' and 'Search interpretation test: text matching 'test''. A table lists the search results.

VRF	Prefix	Order	FQDN	Description
No VRF	+ 1.0.0.0/8			
	+ 1.0.0.0/16			klonk
	1.0.1.0/24			test
	- 1.0.5.0/24			bla bla bla4
	1.0.5.1/24			test host 1
	1.0.5.2/24			test host 2
	1.0.5.3/24			test host 3
	1.0.5.4/24			test host 4
	1.0.5.5/24			test host 5
	1.0.5.6/24			test host 6
	1.0.5.7/24			test host 7
	- 1.3.0.0/16			bla bla
	1.3.0.0/24			test
	1.3.3.0/24			blahona
	2.0.1.0/24			test
	2.0.5.0/24			test
	2.0.6.0/24			test
	2.0.7.0/24			test
	2.0.8.0/24			test

<https://spritelink.github.io/NIPAP/assets/images/screenshots/search.png>

NIPAP CLI

- NIPAP IP Address Management værktøj
<https://spritelink.github.io/NIPAP/>
- Findes et CLI Command Line Interface, nipap-cli
- Kunne man mon koble SNMP sammen med nipap-cli
- Værktøjet er et studie værd, man kan vælge output kolonner, hvilket gør det nemt at viderebearbejde data!

NIPAP CLI Example

```
user@host $ nipap address list --columns vrf_rt,prefix,description,total_addresses,used_addresses,free_addresses
Searching for prefixes in any VRF...
```

VRF	RT	Prefix	Description	Total addresses	Used addresses	Free addresses
-		1.0.0.0/8	foo	16777216	8388608	8388608
-		1.0.0.0/9	foo	8388608	4194304	4194304
-		1.0.0.0/10	foo	4194304	2097152	2097152
-		1.0.0.0/11	foo	2097152	1048576	1048576
-		1.0.0.0/12	foo	1048576	384	1048192
-		1.0.0.128/25	kaka	128	0	128
-		1.3.3.0/24	foo	256	0	256
-		2001:db8:1:1::/64	kaka	18446744073709551616	0	

Custom column output, example from
https://spritelink.gitbooks.io/nipap-user-guide/content/cli/list_prefixes.html

NIPAP example, adding firewall cluster interfaces - multiple VLANs

```
#!/bin/sh
# Add sites to NIPAP, dont waste time doing it with mouse
addsite ()
{
    SITE=$1
    SITEID=$2
    for VLAN in 100 200 300 400 500
    do
        nipap address add family ipv4 prefix 172.$SITEID.$VLAN.1 node $SITE-fw.example.net
        nipap address add family ipv4 prefix 172.$SITEID.$VLAN.2 node $SITE-fw-01.example.net
        nipap address add family ipv4 prefix 172.$SITEID.$VLAN.3 node $SITE-fw-02.example.net
    done
}

addsite dk-odense 1231
addsite dk-svendborg 1232
addsite se-stockholm 1233
addsite se-malmoe 12344
```

Automating saves time, and less errors from manual input!

SNMP walk

- Tilbage til PDU'erne - hvor man kan skrive port description og SNMP er slået til
- <http://cdn.raritan.com/download/px2/version-3.3.10/PDU2-MIB-3.3.10-43736.txt>

```
snmpwalk -v2c -c public 172.16.1.2
```

```
SNMPv2-MIB::sysDescr.0 = STRING: PX2 020200
```

```
SNMPv2-MIB::sysObjectID.0 = OID: SNMPv2-SMI::enterprises.13742.6
```

```
snmpwalk -v2c -c public 172.16.1.2 .1.3.6.1.4.1 | grep site
```

```
SNMPv2-SMI::enterprises.13742.6.3.2.2.1.13.1 = STRING: "site-pdu-rack-a-27"
```

```
SNMPv2-SMI::enterprises.13742.6.3.5.3.1.3.1.1 = STRING: "switch site-dist-sw-02"
```

```
SNMPv2-SMI::enterprises.13742.6.3.5.3.1.3.1.11 = STRING: "router site-mx-01"
```

```
snmpwalk -v2c -c public 172.16.1.2 SNMPv2-SMI::enterprises.13742.6.3.5.3.1.3.1
```

Giver hele tabellen fra den PDU - så nu kan vi smide det data videre :-)



Konklusion

- Find API'er
- Brug API'er
- Skriv API'er
- Brug værktøjer der kan automatiseres nemt
- Brug framework der enabler det!

Så bliver Henrik glad :-D