WISP TRACON

MikroTik Scripting

MikroTik MUM – USA 2015

Brian Horn – WISP TRACON LLC

Scripting

- Provides a method to automate tasks through the use of user defined scripts
  - Run on command
    - Use to configure a complete router or group of parameters
      - i.e. Script purchased to configure firewall rules
  - Run on trigger event
    - System Scheduler
      - Executes script at specific time, after specified interval, or both
    - Traffic Monitor
      - Executes script when interface traffic crosses a given threshold
    - Netwatch
      - Executes script based on state of hosts monitored using pings (ICMP)

Configuration Scripts

- Base Configuration (GoldConfig)
  - Standard configuration which is applied to multiple routers
  - Select parameters are then customized
    - Wireless SSID, Channel, Password, System Password
- Feature Configuration
  - Firewall, Queues, …
- Create Script
  - Configure a router, test configuration, export to script file
- Apply Script
  - Import script file to add a feature
  - To create standard configured router
    - Upgrade to required version of RouterOS
    - System -> Reset Configuration -> No Default Configuration -> Run After Reset (Select config file)
    - Customize

Scripting Examples

- Site Power Monitoring
  - Redundant power solution, running on utility or battery power?
- Backbone Link Utilization
  - Alerts based on link utilization
- Customer Utilization Reports
  - Collect monthly utilization data and reset counters
- Remote Wireless Scan
  - What is happening at a remote site
- Router Configuration Backups
  - Automatically backup and transfer configuration files

Transferring the Results

- Data collected by the router needs to be transferred to the application/server where it is processed or stored
- Methods
  - Email
  - FTP
  - SSH

Email Transfer

- Email tool is a utility that allows the router to send emails
  - Can send messages and/or files
- Example using Gmail
  - Configure client to connect to server

    /tool e-mail

    set address=74.125.28.108 port=587 from=userxyz@gmail.com user=userxyz

    password=abcd130426 start-tls=yes
  - Send message with file attached

    send to=monitor@gmail.com subject="Daily Report - Site MER" body="Router ID and Date" file=backup.rsc

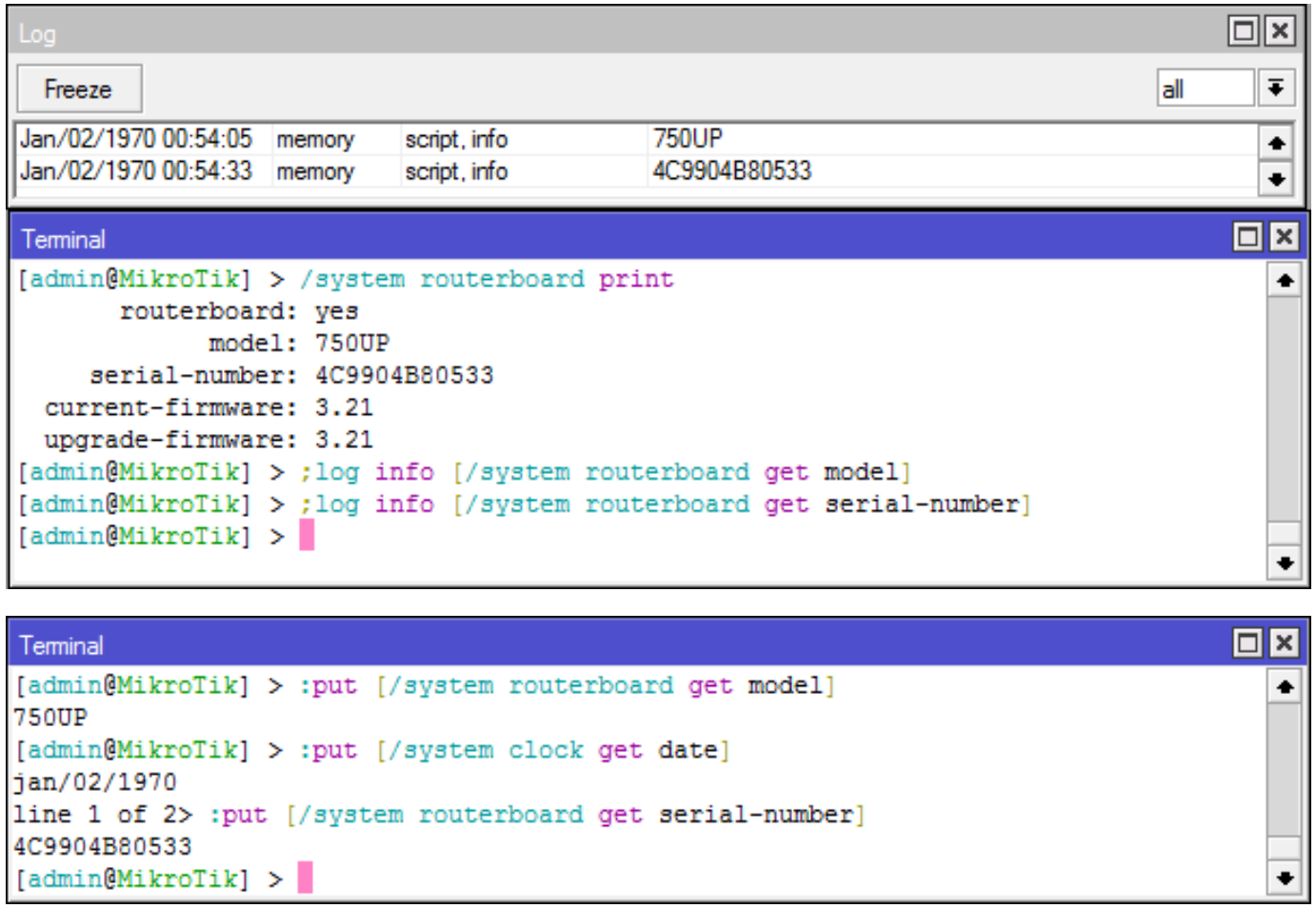
Writing/Testing Scripts

WinBox can provides a simple methods to test commands and see results

Open Log and Terminal windows and use :log

Or

Terminal only and use :put



Getting the System Data

- System Identity

  ```
  :global SystemIdentity [/system identity get name]
  ```

  - Check what you got is correct

    ```
    :put $SystemIdentity
    ```

- System Clock

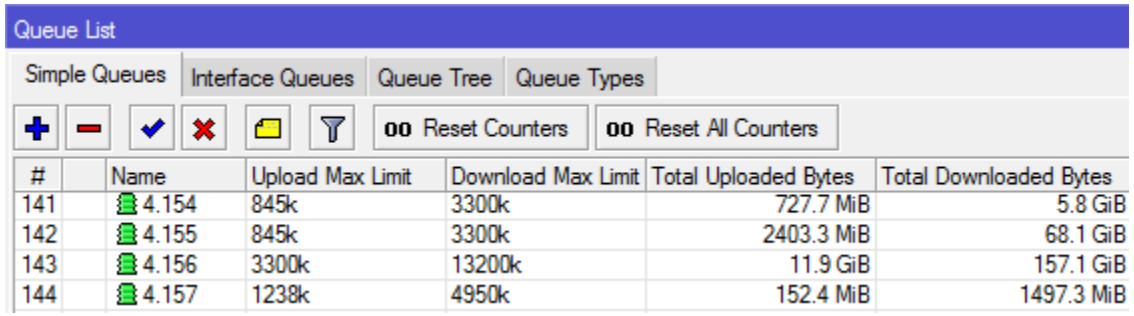
  ```
  :global SystemDate [/system clock get date]
  :put $SystemDate
  ```

  - Extract the day of the month

    ```
    :global Today [:pick [/system clock get date] 4 6]
    :put $Today
    ```

Getting data can be complicated

- Retrieve customer usage statistics and reset counters each month
- /queues simple



#	Name	Upload Max Limit	Download Max Limit	Total Uploaded Bytes	Total Downloaded Bytes
141	4.154	845k	3300k	727.7 MiB	5.8 GiB
142	4.155	845k	3300k	2403.3 MiB	68.1 GiB
143	4.156	3300k	13200k	11.9 GiB	157.1 GiB
144	4.157	1238k	4950k	152.4 MiB	1497.3 MiB

- Script

```
[admin@Data Center - Primary] /queue simple> :for i from=141 to=144 do={:global bytes [g
et $i bytes]; :global name [get $i name]; :put "$name $bytes"}
4.154 763017119/6180455631
4.155 2520032715/73154587562
4.156 12735513834/168677893938
4.157 159566705/1567930340
```

Extracting Fields

- We want to extract Total Upload and Download Bytes

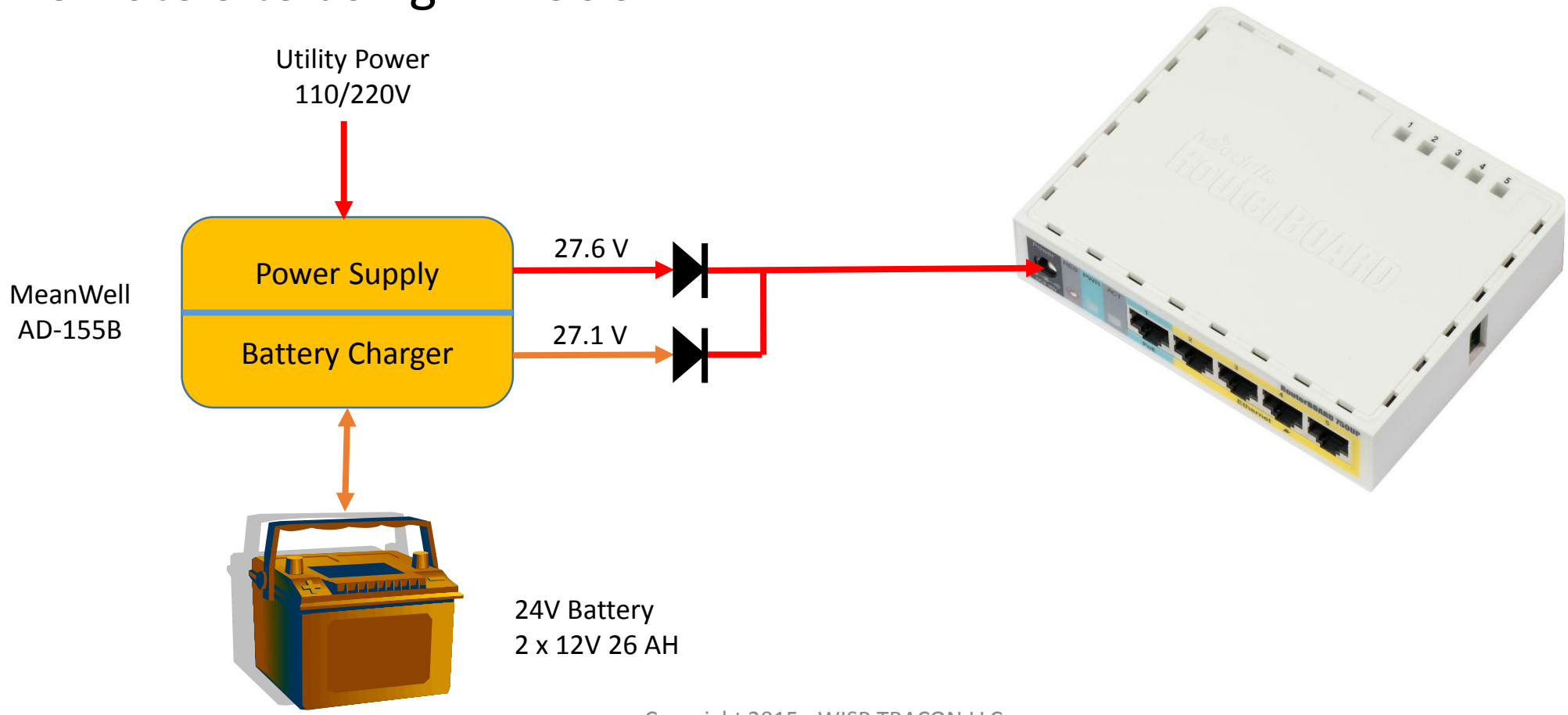
```
[admin@Data Center - Primary] /queue simple> :global bytes [get 141 bytes]
[admin@Data Center - Primary] /queue simple> :put $bytes
802635955/6295490752
[admin@Data Center - Primary] /queue simple> :put [:find $bytes "/" 0]
9
[admin@Data Center - Primary] /queue simple> :put [:pick $bytes  0 9]
802635955
[admin@Data Center - Primary] /queue simple> :put [:len $bytes]
20
[admin@Data Center - Primary] /queue simple> :put [:pick $bytes 10 20]
6295490752
```

- Which you can write as

```
[admin@Data Center - Primary] /queue simple> :put [:pick $bytes 0 [:find $bytes "/" 0]]
802635955
[admin@Data Center - Primary] /queue simple> :put [:pick $bytes ([:find $bytes "/" 0]+1) [:len $bytes]]
6295490752
```

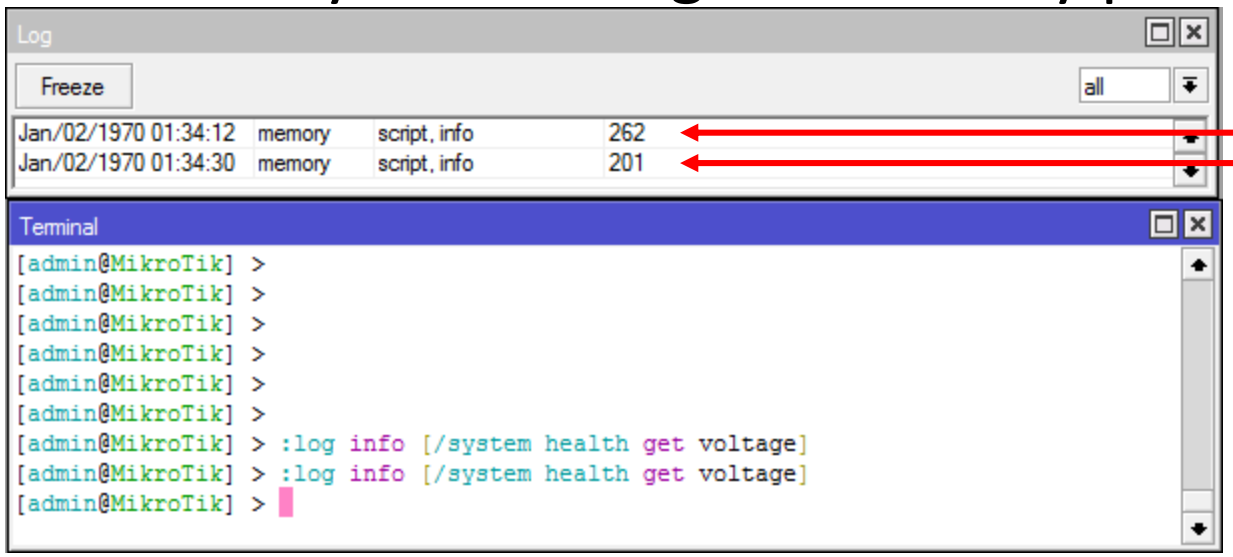
Redundant Power Application

- Remote Site using RB750UP



Redundant Power Control Decisions

- Measure system voltage with utility power on and off



- Determine threshold to use to determine status of utility power
  - Full charged battery ≥ 25.6 V
  - Use 26 V as the threshold

Pseudo Code – Step 1

Measure system voltage (Vsystem)

If running on utility power (Vsystem ≥ 26 V)

    Do nothing

Else (running on battery power)

    Send email to alert power status, include current voltage

What system information do we need for reporting?

Pseudo Code – Step 2

Get system information

- Identity, Date, Time

Measure system voltage (Vsystem)

If running on utility power (Vsystem ≥ 26 V)

- Do nothing

Else (running on battery power)

- Send email to alert power status, include current voltage

Do we want an alert when the utility power is restored?

Pseudo Code – Step 3

Set global variable Vflag = 1

Get system information

    Identity, Date, Time

Measure system voltage (Vsystem)

If running on utility power (Vsystem ≥ 26 V)

Else (running on battery power)

    Clear Vflag (Vflag = 0)

    Send email to alert power status, include current voltage

Pseudo Code – Step 3

Set global variable Vflag = 1

Get system information

    Identity, Date, Time

Measure system voltage (Vsystem)

If running on utility power (Vsystem ≥ 26 V)

    If Vflag not set (Vflag = 0)

        Send email to alert utility power has been restored

        Set Vflag (Vflag = 1)

Else (running on battery power)

    Clear Vflag (Vflag = 0)

    Send email to alert power status, include current voltage

What happens if the batteries get discharged too low?

Pseudo Code – Step 4

```
Set global variable Vflag = 1, Pflag = 0
Get system information
        Identity, Date, Time
Measure system voltage (Vsystem)
If running on utility power (Vsystem ≥ 26 V)
        If Vflag not set (Vflag = 0)
                Send email to advise utility power has been restored
                Set Vflag (Vflag = 1)


Else (running on battery power)
        Clear Vflag (Vflag = 0)
        If battery power > 25% charge (Vsystem ≥ 24 V)
                Send email to alert power status, include current voltage
        Else
                Power down low priority devices
                Send email updating status
                Set Pflag (Pflag = 1)
```

Pseudo Code – Step 4

```
Set global variable Vflag = 1, Pflag = 0
Get system information
        Identity, Date, Time
Measure system voltage (Vsystem)
If running on utility power (Vsystem ≥ 26 V)
        If Vflag not set (Vflag = 0)
                Send email to advise utility power has been restored
                Set Vflag (Vflag = 1)
        If Pflag set (Pflag = 1)
                Power up low priority devices
                Clear Pflag (Pflag = 0)
Else (running on battery power)
        Clear Vflag (Vflag = 0)
        If battery power > 25% charge (Vsystem ≥ 24 V)
                Send email to alert power status, include current voltage
        Else
                Power down low priority devices
                Send email updating status
                Set Pflag (Pflag = 1)
```

Writing the Script 1

```
# Declare variables
:global Rflag
:global Vflag
:global Pflag

# Initialize variables if run for first time
:if ($Rflag!=1){
    :set Rflag 1
    :set Vflag 1
    :set Pflag 0
}

# Get system information
:local Location [/system identity get name]
:local SystemDate [/system clock get date]
:local SystemTime [/system clock get time]

# Get system voltage
:local Vsystem [/System health get voltage]
:local ptr

# Configure email
/tool e-mail set address=74.125.28.108 port=587 from=userxyz@gmail.com \
user=userxyz password=abcd130426 start-tls=yes
```

Writing the Script 2

```
# Main script
:if (Vsystem >= 260) do={
# On utility power
    :if (Vflag=0) do={
# Power outage on last script run
# Send email to advise utility power has been restored
        /tool emaill send to=monitor@gmail.com \
        subject=("Utility Power Restored - $Location ".$SystemDate .$SystemTime) \
        body=("Alert: ".$Location " voltage is ". [:pick $Vsystem 0 2] . "." . [:pick $Vsystem 2 3]."V")
        :log info “Utility power restored at $Location”
        :set Vflag=1;
    }
    :if (Pflag=1) do={
# Power up devices that were shut down
        :for ptr from 2 to 3 step 1 do={
            /interface Ethernet poe set “ether$ptr” poe-out=auto-on
            :log info “Power on port $ptr enabled”
        }
        :set Pflag=0;
    }
```

Writing the Script 3

```
} else={
# On battery power
   :if (Vsystem >=240) do={
# Send email to advise battery voltage is getting low
       /tool email send to=monitor@gmail.com \
       subject=("Low Battery Voltage - $Location ".$SystemDate .$SystemTime) \
       body=("Alert: ".$Location " voltage is ". [:pick $Vsystem 0 2] . "." . [:pick $Vsystem 2 3]."V")
       :log info ("Low battery voltage - ".$Location - " voltage is ". [:pick $Vsystem 0 2] . ". [:pick Vsystem 2 3]."V")
   } else={
# Battery charge < 25% power down low priority devices (ether 2 & 3) & log action
       :for ptr from 2 to 3 step 1 do={
          /interface Ethernet poe set “ether$ptr” poe-out=off
          :log info “Power on port $ptr disabled”
       }
# Send email to advise battery voltage is getting low
       /tool email send to=monitor@gmail.com \
       subject=("Low Battery Voltage – Low Priority Devices Disabled - $Location ".$SystemDate .$SystemTime) \
       body=("Alert: ".$Location " voltage is ". [:pick $Vsystem 0 2] . "." . [:pick $Vsystem 2 3]."V")
       :log info ("Low battery voltage - ".$Location - " voltage is ". [:pick $Vsystem 0 2] . "." . [:pick Vsystem 2 3]."V")
       :set Pflag=1;
   }
}
```

Scripting Tips

- Use pseudo code to structure goal
- Use meaningful variable names
- Comment code
- Test code in sections and debug
  - Global variable value can be set, rather than [get] to check code operation
- Caution when:
  - Using Netwatch and :reboot
  - Disabling or powering down interfaces

The Solutions Training and Consulting Company