

Customer Enabled Monitoring and Control

A Case study:

An Enhanced Managed Service:

- MikroTik routing for data centre, towers and CPE
- Taking advantage of MikroTik's API

2019 - Internet of Things - IoT

- ▶ Internet in the home/office is now not just about streaming and browsing
- ▶ Typical home is Wi-Fi enabled
 - ▶ Up to 5 Wi-Fi access points/repeaters
 - ▶ Customers expect devices to be plug and play
 - ▶ No concept of frequency assignment or interference
 - ▶ Higher powered devices are better!
 - ▶ Higher priced devices must be better!
 - ▶ No concept of firewalls, bandwidth utilization or traffic prioritization
 - ▶ 20 to 40 wireless attached devices
- ▶ Typical home now has IOT devices (cameras, home automation)
 - ▶ Customers have no idea what “port forwarding” means, but IOT devices need it

A Managed Service for Customers

Why did the ISP's in this case study chose end to end MikroTik?

- ▶ One technology and a consistent User Interface across all devices
 - ▶ Minimizes training and support costs
- ▶ Training
 - ▶ Extensive training available
 - ▶ MUM a valuable venue for exchanging ideas and learning
- ▶ Increased profitability
 - ▶ Scripted service set up and infrastructure alarms
 - ▶ Decreased operating expenses - staff and truck rolls
- ▶ Customers have no access to their routers
 - ▶ Consistent customer configurations, minimised configuration errors

A Managed Service for Customers

Why did the ISP's in this case study chose end to end MikroTik?

- ▶ One technology and a consistent User Interface across all devices
 - ▶ Minimizes training and support costs
- ▶ Training
 - ▶ Extensive training available
 - ▶ MUM a valuable venue for exchanging ideas and learning
- ▶ Increased profitability
 - ▶ Scripted service set up and infrastructure alarms
 - ▶ Decreased operating expenses - staff and truck rolls
- ▶ Customers have no access to their routers
 - ▶ Consistent customer configurations, minimised configuration errors
 - ▶ But now in 2019 this is a mixed benefit

Need for an Enhanced Managed Service

- ▶ Problem :Increased customer support issues
 - ▶ 2017: Analysis of technical support calls @ Winters Broadband, North California
 - ▶ Over 90% of calls were related to issues in customer home network
 - ▶ Customers expected support to be provided for their home networks
 - ▶ Technical support was becoming costly and time consuming
- ▶ Solution:
 - ▶ Give customers *some* access to their router?

Need for an Enhanced Managed Service

Giving back some control - the challenge!

- ▶ If customers can see they are getting all of the service they purchased, and they can see who in the home is hogging their service, they can negotiate with them to be more reasonable
 - ▶ and they won't call complaining their service is slow
 - ▶ but they might call to buy more bandwidth
- ▶ If they can see and simply prove that the latest wifi gadget they bought is interfering with their service delivery
 - ▶ they call the gadget vendor for support, not you, the ISP
- ▶ If they can see you delivered the port forwarding they requested for remote camera access/ home automation
 - ▶ they call the camera/home automation supplier for support, not you

Giving back some control - A message from customers..

“ Enhance my service..

- ▶ Show me how much data I'm using and show me you are delivering the bandwidth I paid for
- ▶ Let me judge what's right for me - don't make me pay for a service level I don't need
- ▶ Let me see who's on line and how they are consuming my service
- ▶ When things aren't working with a new device -give me the information I need to get the right support from the device supplier
- ▶ Let me control what the kids can access, *and let me limit their time on line*

...but make it quick and really easy for me to use ”

Giving back some control to customers

ISP Win

Decreased support

Differentiation from
competing ISPs

Customer Win

Customers get what they want

....MikroTik's API can deliver a solution

Solution: MikroTik's wide capability API

- ▶ RouterOS API commands broadly follow the terminal syntax:
 - ▶ All terminal commands are supported by the API
 - ▶ API command word elements must be separated by a "/" not space as with the terminal
 - ▶ EG API command: `/interface/ethernet/print` is equivalent to terminal: `/interface ethernet print`
 - ▶ API commands cannot be abbreviated in the same way as the terminal commands
 - ▶ Eg terminal: `/int eth pr` works, but API: `/int/eth/pr` returns an error
 - ▶ **API supports query words which filter/limit data returned from an API call**

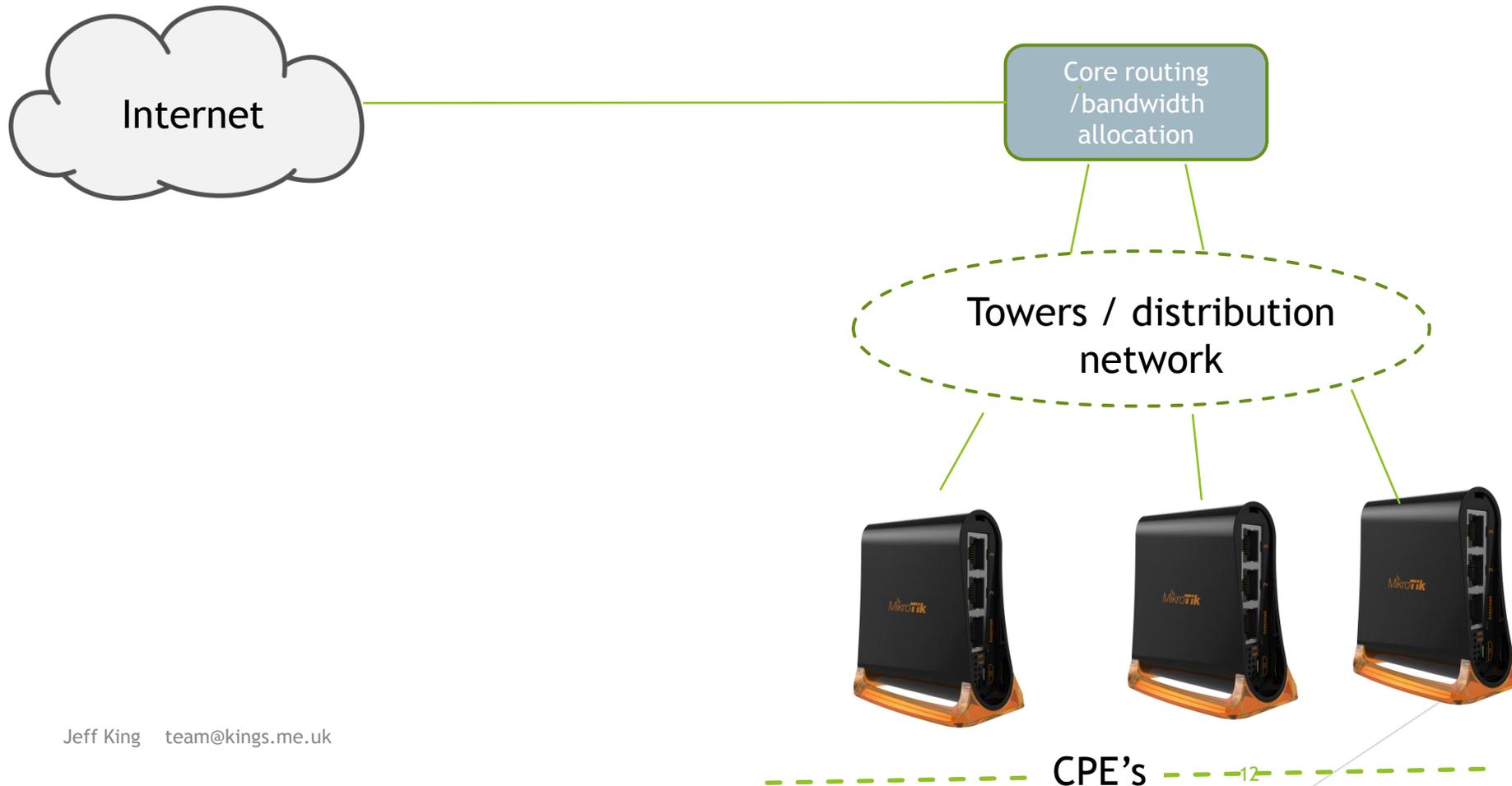
Solution: MikroTik's wide capability API

- ▶ There are API classes for a wide range of languages (python,PHP,C, C#, C++,Ruby on rails,vb etc)
 - ▶ In each case, the API is treated as an object with the interface class handling the low level API protocol. The connection to a specific router is made by passing parameters:
`connect(<router IP address><user name><password>)`
 - ▶ The returned data from an API command is assembled into an array, rather than in the tabular form returned from a terminal command
 - ▶ It is easy to programmatically manipulate data in this form

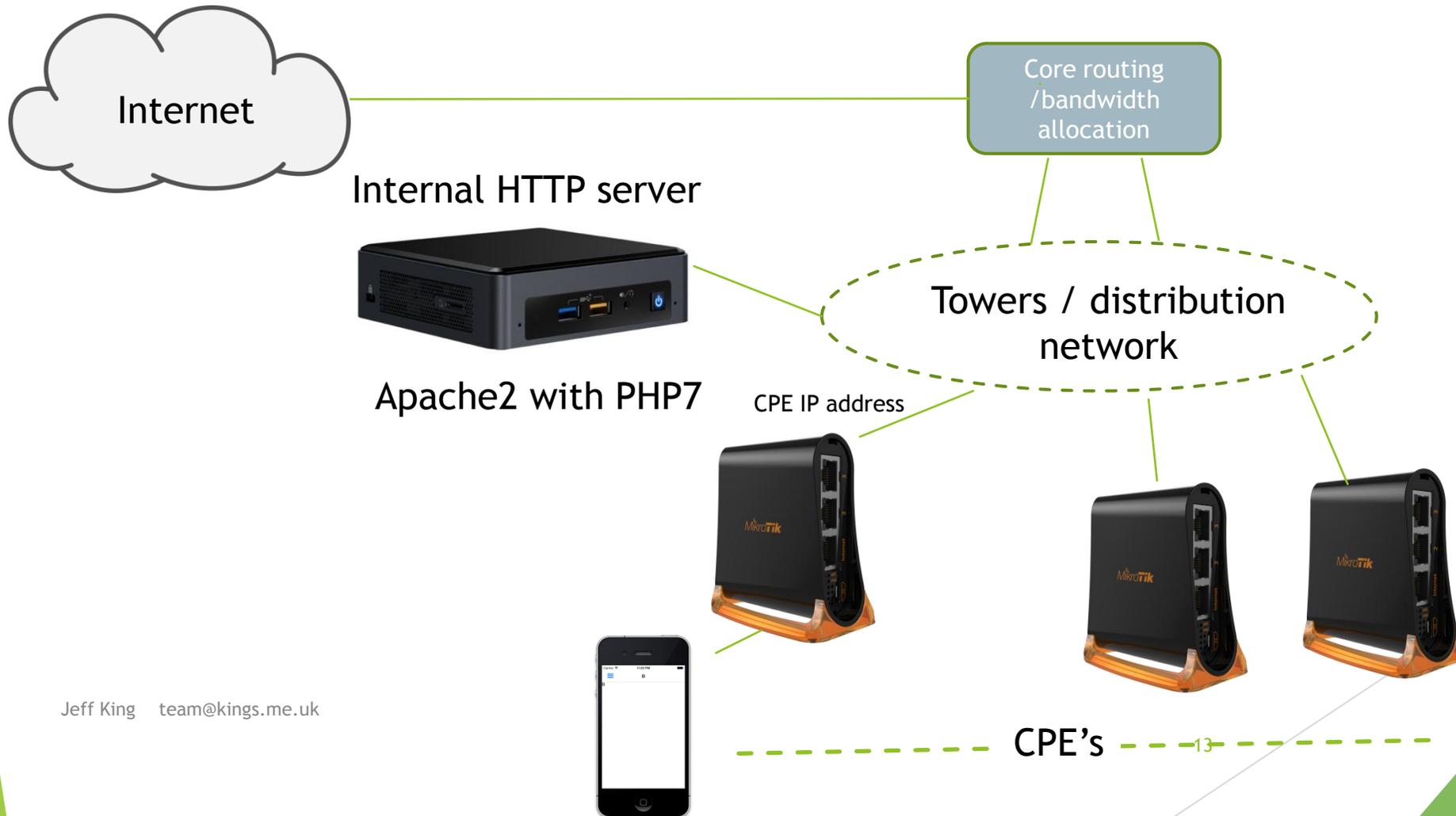
Solution: MikroTik's wide capability API

- ▶ The API is ideal for presenting the limited, specific and ideally simplified monitoring and parental control that ISP customers need
 - ▶ Ability to script information retrieval from two routers (using API connects) and combine the received data
 - ▶ One view we wanted to show involved looking at the customer router and the core network router, so we could create a single dial showing the current download rate and max service limit available to the customer
 - ▶ Ensuring e.g. time, date and time zones are set correctly on a customer router
 - ▶ A server's time and date info can be used to set up a router time and date configuration using API
 - ▶ Setting schedules and scripts on a router for parental control using API
- ▶ The API could be used very effectively for other essential monitoring activities
 - ▶ E.g. Monitoring `/system/health` for battery voltage of remote infrastructure routers
 - ▶ Scripting alarms (via email, sending alerts or setting actions) is easier to set up/ debug with the help of API and e.g. php language tools rather than running scripts on the router itself.

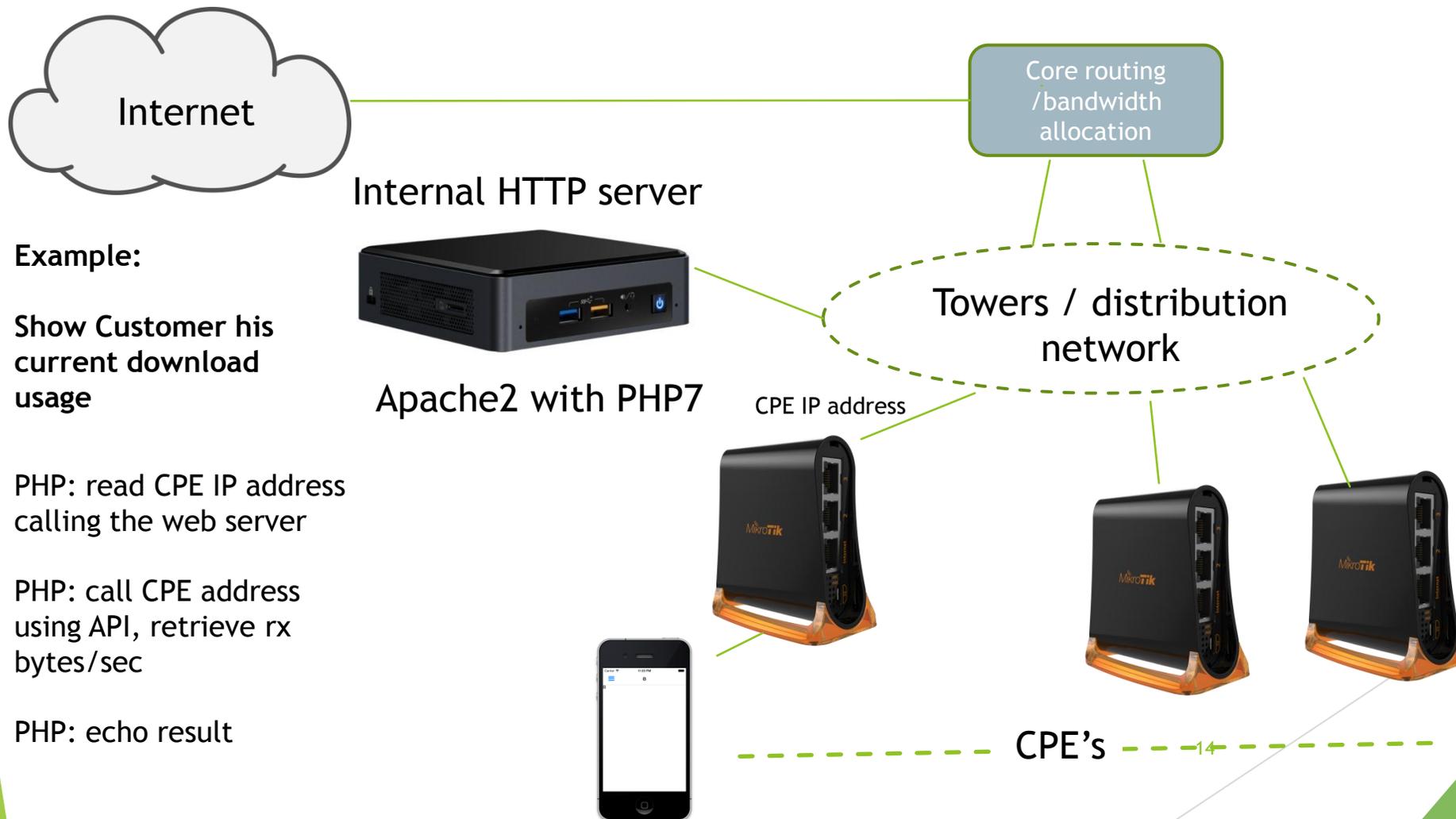
How we used the API to build user monitoring...



Using MikroTik's API to build user monitoring



Using MikroTik's API to build user monitoring



Example:

Show Customer his current download usage

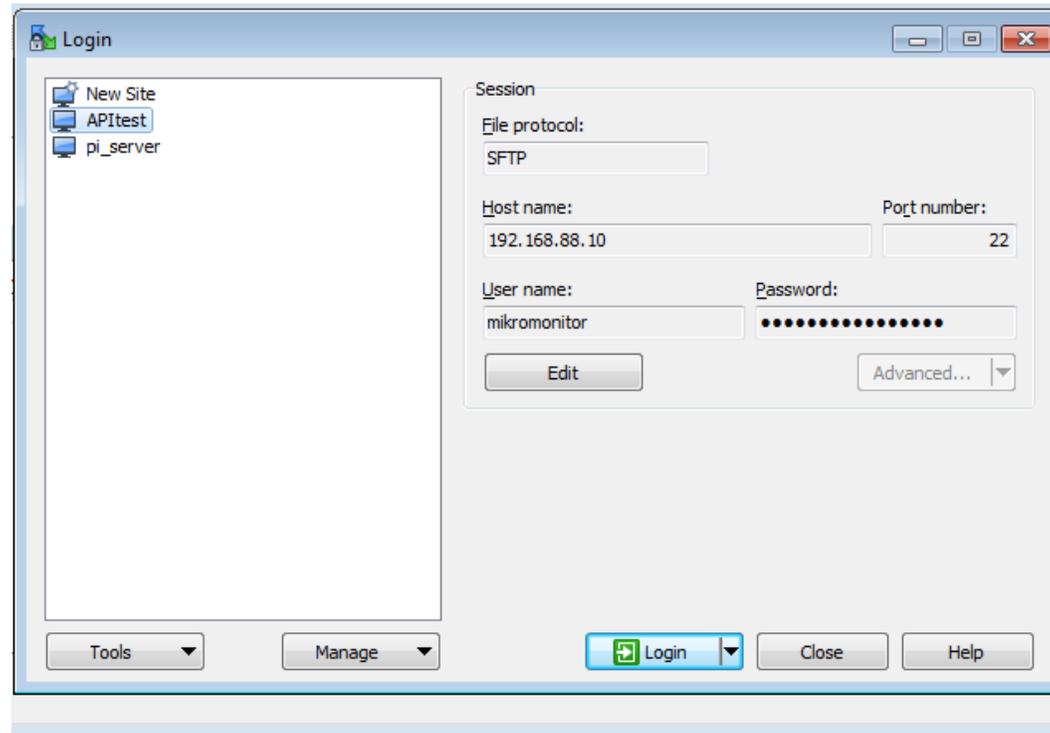
PHP: read CPE IP address calling the web server

PHP: call CPE address using API, retrieve rx bytes/sec

PHP: echo result

PHP API code example

Reading download rate

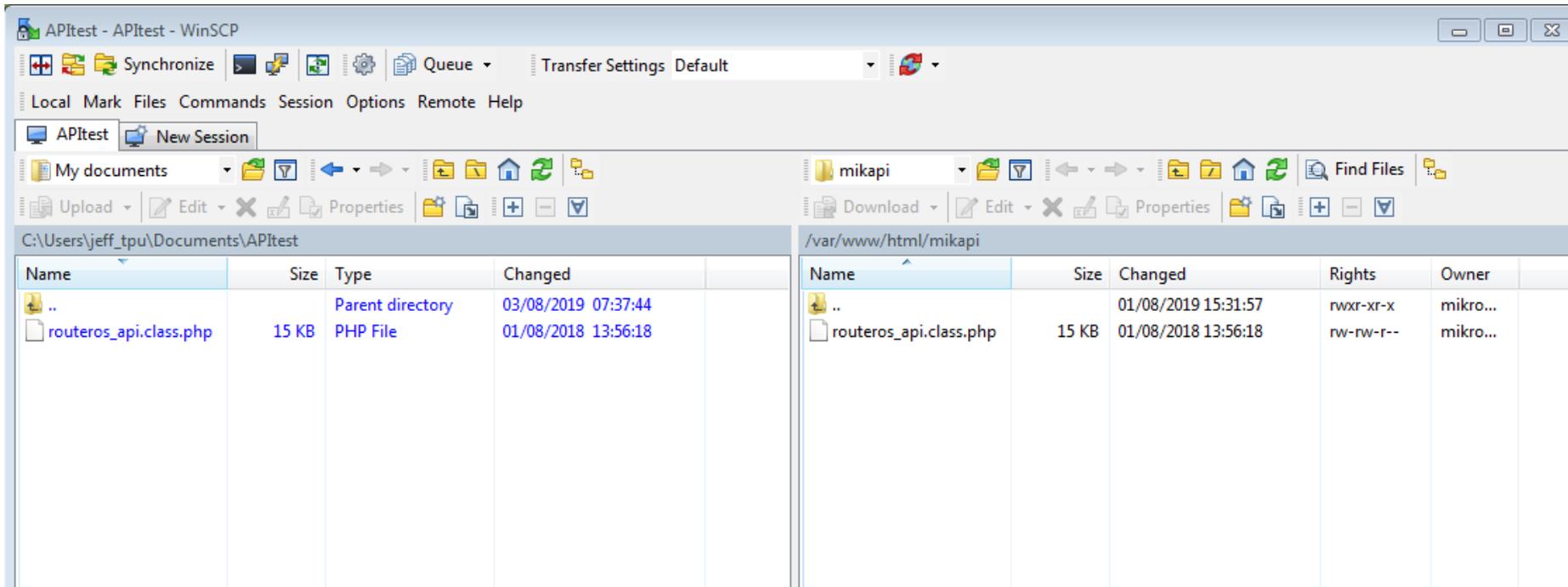


Required: a LAMP (Linux, Apache2,MySQL,PHP) or WAMP (Windows) server

Connect to it from a PC using WinSCP or Filezilla

PHP API code example

Reading download rate



Required: PHP API class

- create object, connect to a router
- write commands to connected object, class executes API communications, returns result in array

Move PHP API class file to Apache web directory on the LAMP server

PHP API code example

Reading download rate

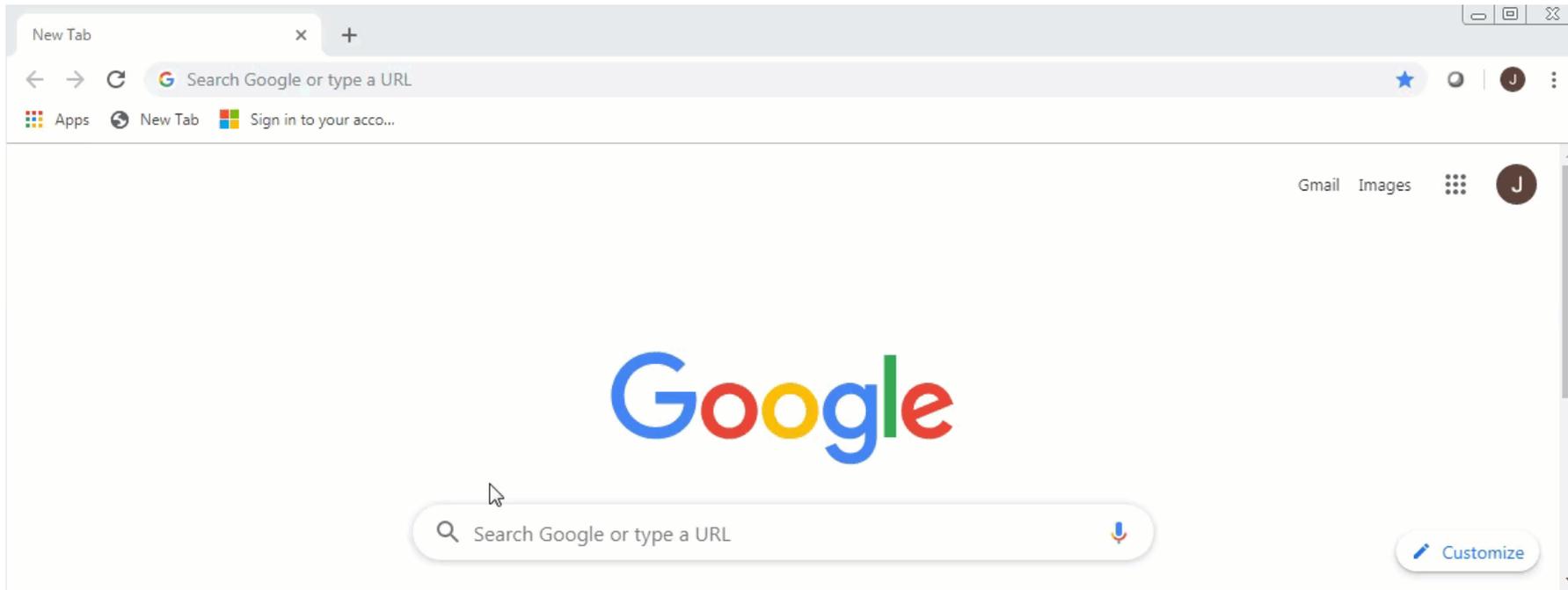
1. Create a file “downrate.php” on the web directory which when accessed by a browser, will show a snapshot of download rate
2. PHP code in downrate.php:
 - a) Include API class, set up an object name to use
 - b) Connect to the router to be managed
 - c) Use API commands to find the router uplink
(the port that is running and is not slave)
 - d) Use API commands to get a snapshot of the port download rate
 - e) “echo” the result to the browser

```
1 <?php
2 require_once('routeros_api.class.php'); //include the API class
3 $API = new RouterosAPI(); //create an API object
4 if ($API->connect('192.168.88.254', 'test', 'test')) { // connect format: (ip of router,user,password)
5     // find the uplink port
6     $API->write('/interface/ethernet/print');
7     $READ = $API->read(false);
8     $ARRAY = $API->parseResponse($READ);
9     $arrlength=count($ARRAY);
10    for($x=0;$x<$arrlength;$x++) {
11        if(!isset($ARRAY[$x]['slave']) and $ARRAY[$x]['running']=="true") { $name=$ARRAY[$x]['name'];}
12    }
13    // now get a snapshot of download (receive traffic rate)
14    $API->write('/interface/monitor-traffic',false);
15    $API->write('=interface='.$name,false);
16    $API->write('=once=',true);
17    $READ = $API->read(false);
18    $ARRAY = $API->parseResponse($READ);
19    echo $ARRAY[0]['rx-bits-per-second']; // send just the receive rate from array
20 }
21 else {echo "could not connect";} // report failed connect
22 ?>
```

PHP API code example

Reading download rate

► Run the code:



Using Javascript gauges to show rates

Google Charts

HOME

GUIDES

REFERENCE

SUPPORT

Overview

Hello, Charts!

Quickstart

Load the Charts Library

Prepare the Data

Customize the Chart

Draw the Chart

Draw Multiple Charts

Chart Types

Chart Gallery

Annotation Charts

Area Charts

Bar Charts

Bubble Charts

Calendar Charts

Candlestick Charts

Column Charts

Combo Charts

Visualization: Gauge

Overview

A gauge with a dial, rendered within the browser using SVG or VML.

Example

Computational resources



JqueryAJAX code example: updating Google charts gauge using Javascript, PHP and API

- ▶ Use Javascript in the web page to continuously poll download rate from a router
- 1. Create an index.html file on the web directory which when accessed by a browser, will display a moving download rate gauge
- 2. PHP and javascript code
 - In header section of page:
 - a) Load JQuery javascript tools
 - b) Extend JQuery with an AJAX function to get data from our downrate.php file and pass it through to Javascript code running in the accessing browser
 - c) Load google gauges javascript code
 - In body of page:
 - d) Create a chart function using google gauges template, get data from the downrate.php file to draw gauge.
 - e) Include a 'SetInterval' function to redraw chart every second.

Page header code

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" >
<head>
<title>Download</title>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
<script>
jQuery.extend({
  getValues: function(url) {
    var result = null;
    $.ajax({ url: url,
      type: 'get',
      dataType: 'html',
      async: false,
      success: function(data) {
        result = data;
      }
    });
    return result;
  }
});
</script>
<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
</head>
```

```

<body>
<script type="text/javascript">
  google.charts.load('current', {'packages':['gauge']});
  google.charts.setOnLoadCallback(drawChart);
  function drawChart() {
    var rates=($.getValues("downrate.php))/1000000;
    var data = google.visualization.arrayToDataTable([
      ['Label', 'Value'],
      ['Download', rates],
    ]);
    var options = {
      width: 300, height: 300,
      minorTicks: 3,
      majorTicks: ['0','5','10','15'],
      max:15
    };
    var chart = new google.visualization.Gauge(document.getElementById('chart_div'));
    chart.draw(data, options);
    setInterval(function() {
      var rates=($.getValues("downrate.php))/1000000;
      data.setValue(0, 1, rates);
      chart.draw(data, options);
    }, 1000);
  }
</script>
<p>Download rate:</p>
<div id="chart_div"></div>
</body>
</html>

```

Load gauge

Start chart draw function when page is loaded

Function to draw chart

get a download rate reading

set data in gauge table

Set chart options (size, scale)

Create chart object

Draw chart

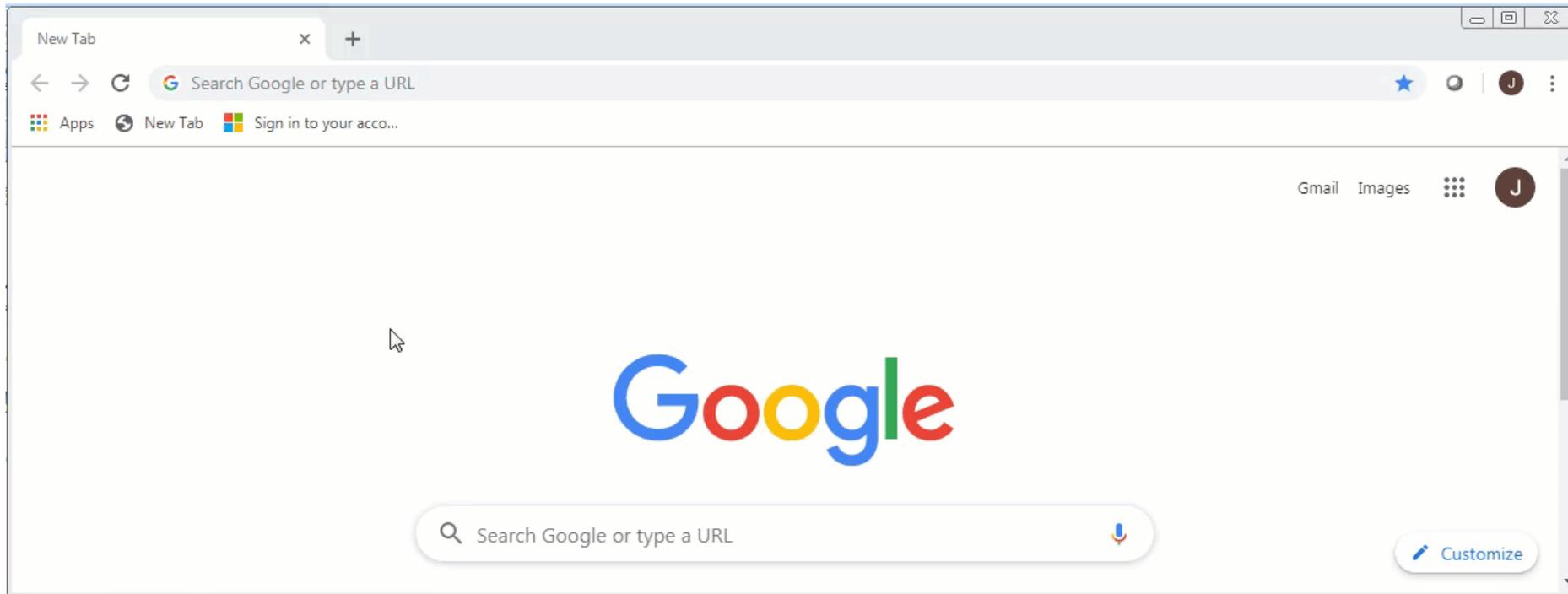
Set a chart repeat interval

...to redraw chart

Place in page where chart will display

JqueryAJAX code example: updating Google charts gauge using Javascript, PHP and API

► Run the code:



Case Study: The Screens we presented

▶ Bandwidth

- ▶ What service level have I subscribed to?
- ▶ How much bandwidth am I using?
- ▶ Who is using it?

Current Top Bandwidth User(s)

Traffic Direction	User Name	Bandwidth Use
Download	DESKTOP-3TE2UCB	0.65Mbps
Upload	DESKTOP-3TE2UCB	< 0.1Mbps

Managed Router Monitor
John Smith

Monday 12 November 2018 14:27:59

Your IP address is 208.91.7.211

Technical Support:

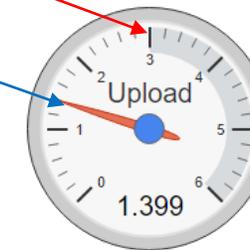
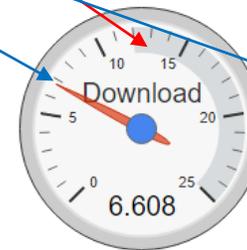
530-852-7990

support@winters-broadband.com

Accounts:

530-852-7990

accounts@winters-broadband.com



Case Study: The Screens we presented

► Network

- What is my public IP address?
- Who is connected to my home network?
- Why is the service to some of my devices slow?

Managed Router Monitor
John Smith
Monday 12 November 2018 14:27:59
Your IP address is 208.91.7.211
Technical Support:
530-852-7990
support@winters-broadband.com
Accounts:
530-852-7990
accounts@winters-broadband.com

Connected Devices

Device Name	IP Address	On Wi-Fi?	Signal Strength
undefined	192.168.88.207	No	
ABB1000	192.168.88.208	2G WiFi	●●●
DESKTOP3TE2UCB	192.168.88.188	No	
Bedroom	192.168.88.204	No	
HPB6C38A	192.168.88.252	2G WiFi	●
LivingRoom	192.168.88.229	2G WiFi	●●●
MarksiPadAir	192.168.88.243	2G WiFi	●
FamilyRmAppleTV	192.168.88.237	2G WiFi	●●

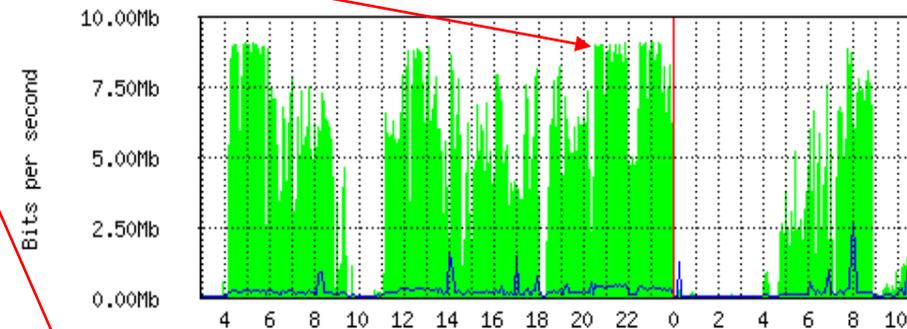
Case Study: The Screens we presented

► Usage

- How much data have I used this month?
- Why was my service slow at 9 pm last night?

Download and Upload Usage

One Day History Graph (5 Minute Average)



Monthly Usage

Download: 1,112.26 GB Upload: 44.91 GB

Case Study: The Screens we presented

- ▶ Port Forwarding
 - ▶ What ports are forwarded to my security system?

Ports Forwarded

Protocol	Port	Forwarded to
tcp	9010	192.168.88.252
tcp	81	192.168.88.252
udp	9010	192.168.88.252

Case Study: The Screens we presented

- ▶ Content Filtering

- ▶ Use API to configure DNS target for specific customer router users
- ▶ Point user at (freely available) OpenDNS Family Shield addresses, which block adult content

- ▶ Presented Customer Access Control Screen

- ▶ Allow customer (Mum/Dad) to select a specific device from a list of their connected devices
- ▶ We set up password access to this screen so that only Mum or Dad could set it

Name	Content Filtering	Parental Control Profile	Interfaces
DESKTOP-157IVLU	OFF ▾	OFF ▾	1

Case Study: The Screens we presented

- ▶ Screen presented to Mum/Dad to set up profiles that can be applied to any device
 - ▶ Children need to go to sleep!
 - ▶ Children need to do homework!

Overnight Blockers	Start	Stop
Weeknights	9pm ▼	7am ▼
Weekends	10pm ▼	6am ▼

Daytime Blockers	Start	Stop
Monday	▼	▼
Tuesday	4pm ▼	6pm ▼
Wednesday	▼	▼
Thursday	▼	▼
Friday	▼	▼
Saturday	▼	▼
Sunday	▼	▼

Case Study:

Giving back (limited) control-What was the impact?

- ▶ Cutting support time and costs
 - ▶ A marked reduction in customer calls- the effect of customer self help
 - ▶ Quicker resolution of customer issues- use of the screens during customer calls
- ▶ Improved quality of service
 - ▶ Interestingly, potential customer issues were proactively resolved during screen rollouts
- ▶ Smooth transition of aging (non MikroTik) CPE routers to MikroTik as they are retired
 - ▶ We used the API to script new router set up
 - ▶ Availability of monitoring and access control screens made a concrete case for customers to pay for a router upgrade
- ▶ Revenue gains: customers encouraged to upgrade service, loyalty, gaining new customers

Case Study:

Giving back (limited) control-What was the impact?

- ▶ We turned all of the CPE base of hap ac lite's and hap mini's into SMART WiFi routers!

www.mikromonitor.com

Case Study:

Giving back (limited) control-What was the impact?

▶ Customer quotes:

- ▶ “We saw that a device we have not used in ages was consuming most of our download bandwidth all the time...we did not know it was still on line!”

“After we switched it off our service greatly improved”
- ▶ “We really needed to stop our teenagers gaming on line half the night. A profile was easy to set up and apply”
- ▶ “Content filtering was important for us - it was simple to set up on the router”
- ▶ “For security reasons and to ensure our service was responsive for our main business users, it was important for us to be able to audit who was connected and active on our service”