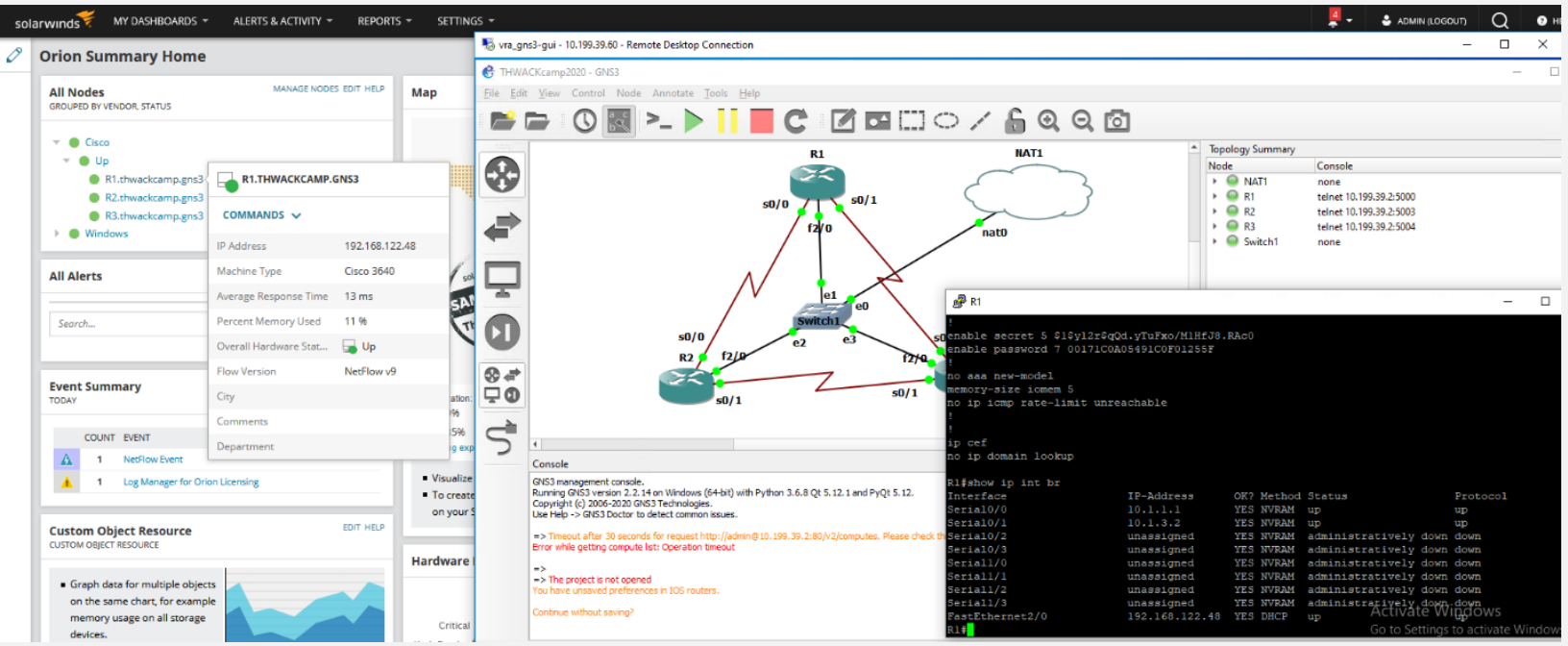
solarwinds



A Virtualized Monitoring Lab With SolarWinds and GNS3

It's All in Your Mind (or On Your Laptop. Or in the Cloud!)

By Leon Adato

Table of Contents

Acknowledgements

It's self-evident (or should be, once you glance at a few pages) this book wouldn't exist without the innovative thinking and old-fashioned hard work of hundreds of people. For starters, there's an entire team of folks who've made GNS3 a reality; and likewise, thousands who've had a hand over the past 20 years in making SolarWinds a trusted name in monitoring solutions.

Of course, those teams include gifted and dedicated designers, managers, engineers, developers, testers, and more. But, for both GNS3 and SolarWinds, another group is conspicuous in both their size and influence: the GNS3 and SolarWinds user communities. Both are far more than a simple support forum or user group. Both embody the best of what the word "community" means—providing unvarnished opinions, unambiguous feedback, and (through it all) unwavering support. For both GNS3 and SolarWinds, the critical ingredient to success has been the people who gather in those forums to help each other out and cheer each other on in equal measure.

It's less evident, and therefore more important for me to acknowledge, the assistance of a few folks who helped me overcoming the very specific challenge of accessing GNS3 devices from a separate instance of SolarWinds monitoring. Without their help, this eBook could not have come into being.

Mark Blackwell—the driving personality behind GNS3, who helped marshal the resources I needed to get testing under way, and who believed in my vision for this project from the beginning back in 2014.

Jeremy Grossmann—who spent several sleep-deprived nights across time zones to plumb the depths of user permissions, installation routines, sub-networking assumptions, and the black magic that is iptables.

Scott Pickles—a "mere" volunteer on the GNS3 contributor forums, who has sacrificed sleep, food, and time with his family to help noodle through some of the toughest conundrums along the way; and whose enthusiasm for results often eclipsed mine.

"Thank you" doesn't even begin to cover the gratitude I feel, but I'm not sure any amount of words would ever suffice.

Dedication

To Debbie

The plain truth is eBooks like these are free to everyone except you. You're the one who pays for my effort in the hours I am absent; in the meals you deliver to my desk; and in the countless times you patiently act as "my duck" while I recount my latest technical frustration, only to dash off when I stumble on a solution. I can only hope I am half the friend to you that you have always been to me, and that I'll ever be able to fully repay your emotional labor once the writing is done. I love you F. E. and A., Best Beloved.

Introduction

About This Guide

Did you ever wish you had a lab to test your monitoring, but didn't have "spare gear" just lying around? This guide provides step-by-step instructions to set up a virtual network in GNS3®, and then connect it to SolarWinds® Orion® Platform monitoring for a "virtually" free (or at least low-cost) lab environment.

About GNS3 (i.e., "Shameless Self-Promotion, Part I")

GNS3 is a tool designed to create virtual network devices which act like real network devices. Why would you want to do this? GNS3 is useful for people who want to pass their network certification exams (and don't want to drop thousands of dollars on actual network gear) and creative-but-frugal network professionals who want to mock up and test out their network designs before rolling it into real production.

About SolarWinds (i.e., "Shameless Self-Promotion, Part II")

The author has used SolarWinds tools since 2003, and here's what he's learned since:

First, SolarWinds is Geekbuilt.® Meaning geeks, including SysAdmins, engineers, and other IT professionals produce solutions for other geeks. SolarWinds addresses real problems geeks face every day at work. We don't design solutions based on which buzzwords are getting the most play on social media. Instead, we spend a lot of time talking to people in the trenches to find out not only what they're thinking about in terms of problems, but also how they'd like to see those problems addressed. Their feedback becomes the list of features we build into the next version.

Second, it's modular. You don't need to get the whole suite in one monolithic installation. You can determine which functionality you need, and then get the modules to meet those needs. The modules snap together under a common framework, and also integrate well with solutions from other vendors. Because real geeks know you don't get to pick every single piece of software the company uses, and like a good mutt, heterogenous solutions are often the most robust and faithful allies you can have in the data center. The flipside of this is each module is flexible. Each tool has a variety of "outside-the-box" actions you can take to get almost any job done.

Finally, and there's no way to dress this up, SolarWinds solutions are affordably priced. Especially when you consider the features in each module. Head over to solarwinds.com for more detailed information on products and pricing; or to download a free, unlimited (meaning you can load up as many devices as you want), 30-day demo of any (or all) of the SolarWinds modules.

Pro tip: there are also about two dozen free tools you can download at http://www.solarwinds.com/free-tools/.

GNS3 + SolarWinds = Awesome

With a relatively recent update—the ability to connect to resources and devices outside the virtual sandbox of GNS3 itself—a whole new class of IT pro has a reason to be really excited. Monitoring engineers who want to test out new devices, applications, monitors, reports, alerts, and so on can now set up an entire "fake" network and then test out monitoring against it, which could include servers, routers, switches, and more.

About This Updated Guide

Back in 2014, I wrote the first version of this guide because GNS3 had just come out with the ability to run VMs (i.e., servers) inside the same virtual space where the virtual routers, switches, and other network devices were doing their thing. And it worked… assuming you had the hardware to run everything. Let's face it, the Orion Platform isn't exactly a svelte application. Nor is the SQL database it depends on. And if the network design was on the larger side… well, you're talking about a server-class machine just to run your "little" virtual lab.

Nevertheless, it both proved a point and served a purpose for a substantial group of IT pros.
But 6 years later, I felt I needed to take another look at this modest solution and bring it into the cloud era. This was aided by the fact GNS3 hadn't stagnated during the intervening time. Now, in addition to being able to run virtual

machines (VMs) inside the GNS3 virtual space, you could also connect to the virtual devices from outside the GNS3 framework. Most GNS3 users limited this ability to connecting via telnet or ssh into their virtual routers. But for a monitoring engineer like me, it meant a whole world of possibilities.

When you coupled this capability with the availability of (relatively) cheap, pay-as-you-go resources in the cloud—not to mention point-and-click convenience of SolarWinds solutions on the major cloud provider's marketplace/store catalogue, and I realized a lot of the barriers to usage had been overcome.

Back in 2014, I wrote,
"The challenge was that monitoring engineers often were unfamiliar with setting up networks (in GNS3 or otherwise); and GNS3 users were unfamiliar with the conventions of monitoring solutions like SolarWinds."

That thought is truer today than it was back then. Which is where this guide comes in.

This document is going to provide step-by-step, command-by-command, show-me-with-pictures instructions for installing GNS3, creating a virtual network, installing SolarWinds tools like Network Performance Monitor, Network Configuration Manager, NetFlow Traffic Analyzer, and Server & Application Monitor, and monitoring those GNS3 devices. We're assuming (almost) nothing about what you, the reader, know and providing detailed instructions for all of it.

So, if you're a GNS3 guru and can set up a hybrid OSFP-BGP-EIGRP-RIP network before your morning coffee, you probably can skip ahead to the NPM part. And if you're a veteran SolarWinds expert who has installed NPM so many times you have the screens memorized, you can probably stop once you get your network installed.

But for most people reading this, who likely fall somewhere in the middle, I hope this guide helps you to get to the part that's not just useful, but also valuable: testing out your network and/or monitoring changes in a safe environment before rolling them to production.

What's New?

The original eBook assumed you were going to build everything on personal computer, server, or VM. At the time, GNS3 hosted both the virtual network devices and the (virtual) Orion server. If that's what you want to do, then the 2014 guide is all you need. You can find it here.

In the intervening years, I recognized how cloud—whether public or private—was not only an option, it was the preferable option for getting all this done. First, because so much of the work of setting up servers and applications acquired a "three-clicks-and-you're-done" level of simplicity. Second, because it was a flexible platform in terms of compute resources. And third, because cloud is becoming the preferred platform to rapidly set up (and tear down) lab environments.

This presented a technical challenge. In the original design, everything sat within the same virtual environment: the routers and the Orion server were all in the same "box," so there was no problem with communication, ports, routing, etc. But in moving to a cloud-style virtual environment, the GNS3 GUI, the GNS3 server, and the Orion server were all de-coupled and now communication—especially between the Orion polling engine and the devices running inside GNS3—had to be sorted out.

Long story short: we did it. That's what you'll be reading about here.

Another change we made to this guide was to separate the "big" installations (GNS3 and the Orion Platform) into their own appendices. While it gives this guide a bit of a choose-your-own-adventure feeling, I made the choice because it made updates to this eBook more modular and therefore easier to update without requiring a complete re-write. Paying down technical debt ain't just for coders, folks.

Section 01: Installation and Configuration

Before You Get Started

Before you proceed on to the technical goodies in this eBook, make sure you have the following four virtual machines ready to go:

1) For the Orion polling engine: One VM running *at least* Windows 2016 (it can be a newer version if you want), all patched up to the latest version.
2) For the Orion database: Another* VM running *at least* Windows 2016, all patched up to the latest version and running Microsoft SQL 2016 or later. Again, all patched up to the most recent versions.
3) For the GNS3 GUI: A VM running Windows 10.
    a. This could also be a server if you wanted. But it's not necessary.
4) For the GNS3 server: One VM running Ubuntu 18.04 or later.**

All these VMs need to be on the same subnet.

I'm going to offer some very loose hardware guidelines right now. I do this partially because the amount of compute power you will need is highly dependent on the type of lab you are building. Running SolarWinds NPM, NCM, NTA, SAM, and WPM against a GNS3 design with 30 routers, 10 switches, and five firewalls is going to require significantly more firepower than the simple design we're using in this guide.

With this in mind, let me offer you what we used for this eBook:

1) Orion polling engine: 8 3.12Ghz CPU, 12Gb RAM, 200Gb disk.
2) Orion database: 8 3.12Ghz CPU, 12Gb RAM, 200Gb disk
3) GNS3 GUI: 4 3.12 Ghz CPU, 8Gb RAM, 60Gb disk
4) GNS3 server: 4 3.12 Ghz CPU, 8Gb RAM, 60Gb disk

As with any lab environment, caveat build-er, your mileage may vary, and (as my grandmother always said) "salt to taste."

Overview of Installation Steps

Here's the overview of what we're going to do:

1. Download everything you need
2. Note your IP addresses
3. Set up the GNS3 server
4. Set up the GNS3 GUI
5. Set up the Orion server

* If this is a very small or very short-term lab, you could use SQLExpress on the same server as the Orion poller. But such a configuration isn't meant for any type of robust operations, so please don't imagine it will work well or in production. You have been warned.

** Yes, GNS3 server can run on Windows (and on MacOS, for that matter). But some of the things we need to do for this specific use case are best done (or in some cases, only possible) with Linux. So here we are. If this is your first experience with Linux, let me warmly welcome you to Club Penguin. Let me also reassure you: this really won't require a whole lot of specialized knowledge.

Step 1: Download Everything You Need

While these installers and files go on the various VMs mentioned earlier, it's easier to have everything downloaded in one place first, and then copy the ones you need on a machine-by-machine basis.

- Orion installer - https://www.solarwinds.com/network-performance-monitor
- GNS3 GUI software - https://community.gns3.com/community/software/download/
- SSH/Telnet client - https://www.solarwinds.com/free-tools/solar-putty
- An IOS image for your network device(s)*

There are a few other 100% free downloads you could grab now, which might come in handy later (although they aren't strictly necessary for the proof-of-concept we're describing in this guide).

- NetFlow Flow Tool Bundle - https://www.solarwinds.com/free-tools/flow-tool-bundle
  Distribute, test, and configure flow traffic
- Network Configuration Generator - https://www.solarwinds.com/free-tools/network-config-generator
  Automatically create template-based configuration scripts for network devices
- Subnet Calculator - https://www.solarwinds.com/free-tools/advanced-subnet-calculator
  Ensure your IP addresses don't conflict with one another
- Cost Calculator for Azure - https://www.solarwinds.com/free-tools/cost-calculator-azure
  Monitor your cloud resource costs

Step 2: Note Your IP Addresses

The rest of this guide is going to have some specific instructions regarding the IPs and subnets of the various devices, so it's worth taking a moment now to write them down. If networking is not your strong suit, hopefully someone on your organization's network team can assist you. If not, I strongly recommend reaching out to the community on THWACK.com. We're all friendly folks and we love to help.

- Subnet for all servers: _____
  - I'll be referring to this as "<server-subnet>" for the rest of this book
- You'll also need the subnet mask: _____
  - I'll be referring to this as "<subnet-mask>" for the rest of this book
- Orion polling engine IP address: _____
  - I'll be referring to this as "<Orion-engine>" for the rest of this book
- Orion database IP address: _____
  - I'll be referring to this as "<Orion-db>" for the rest of this book
- GNS3 GUI IP address: _____
  - I'll be referring to this as "<GNS3-GUI>" for the rest of this book
- GNS3 server IP address: _____
  - I'll be referring to this as "<GNS3-server>" for the rest of this book

* Let's have a brief chat about IOS images. The heart of GNS3 is its ability to mimic—from the command line behavior down to the way data is passed—a real network device. Much of this capability is found in the "operating system" (OS) running on those devices. As such, the GNS3 engineers had a choice: reverse every network device on the planet, incurring the ire of hardware vendors (not to mention a number of tediously annoying lawsuits, cease-and-desist orders, and "you're a poopy head" emails from CEOs), or to use an existing OS file. They decided most people typically have a real network, so the OS files could be copied from existing devices. We realize this isn't always the case, but there's not much to do about it. (Why not give out the IOS along with GNS3? We refer you back to those tediously annoying lawsuits, cease-and-desist letters, etc. etc.). I can suggest you check out the GNS3 community (https://gns3.com/community/) for ideas, suggestions, and emotional support.

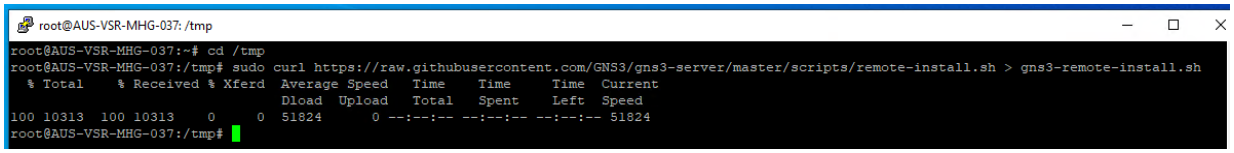
Step 3: Set Up the GNS3 Server

1) SSH to the <GNS3-Server>
2) Install GNS3

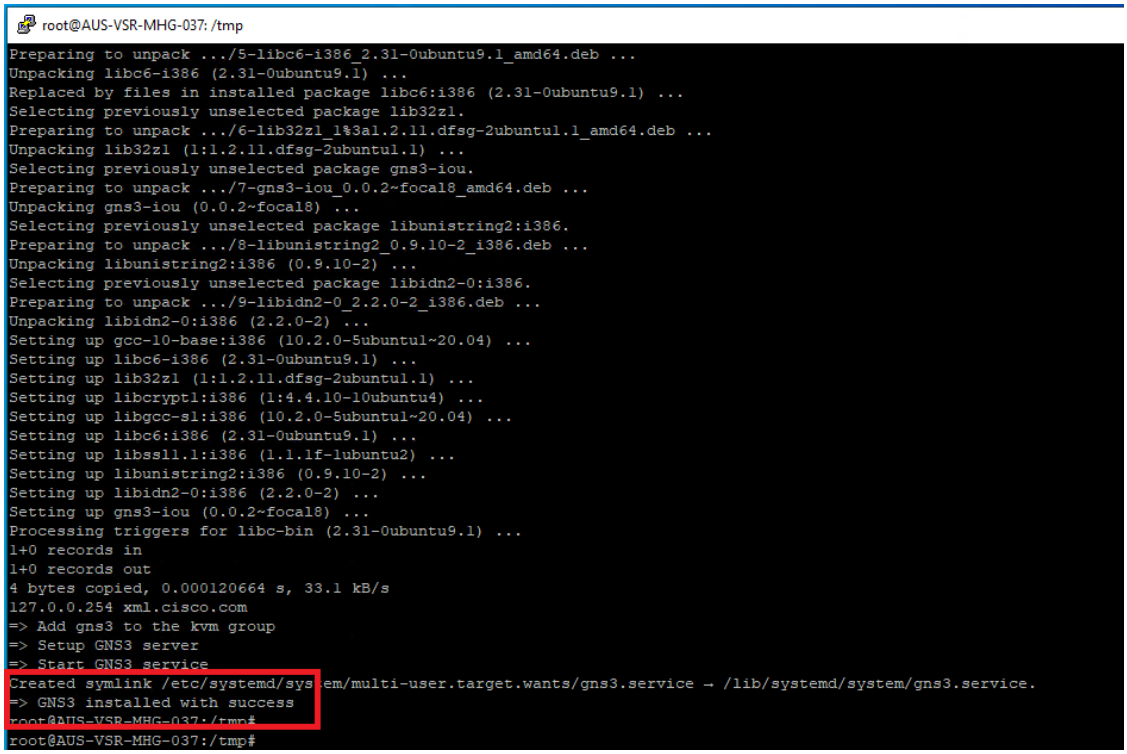
We're going to do this using some specialized scripts rather than installing from pre-build installers. For this reason, the GNS3 Server machine needs to have access to the internet. You can find detailed instructions on this process here: https://docs.gns3.com/docs/getting-started/installation/remote-server/

NOTE: GNS3 only installs on long-term, stable versions of Linux distributions. So, Ubuntu LTS 20.04, for example. You may get an error to this effect, in which case you'll need to upgrade (or, unfortunately, downgrade).

3) Move to the "tmp" directory
   `cd /tmp`
4) Pull the installer script from the GitHub repository
   `sudo curl https://raw.githubusercontent.com/GNS3/gns3-server/master/scripts/remote-install.sh > gns3-remote-install.sh`



5) Execute the installer
   `sudo bash gns3-remote-install.sh—with-iou—with-i386-repository`
   a. Once it's done, you should see a message telling you the installation ran successfully



6) Reboot the machine. GNS3 will be running on the server's IP on port 3080.
   a. You can verify the port by checking /etc/gns3/gns3_server.conf
7) Add a few lines to allow monitoring traffic
```
sudo iptables -I FORWARD -o virbr0 -p icmp—icmp-type echo-request -j ACCEPT
sudo iptables -I FORWARD -o virbr0 -p udp—dport 161 -j ACCEPT
sudo iptables -t nat -I POSTROUTING -s 192.168.122.0/24 -p udp—dport 162 -j ACCEPT
sudo iptables -I FORWARD -o virbr0 -p tcp—dport 22 -j ACCEPT
sudo iptables -I FORWARD -o virbr0 -p tcp—dport 23 -j ACCEPT
```
8) If you plan to do NetFlow monitoring, also add this line:
```
sudo iptables -t nat -I POSTROUTING -s 192.168.122.0/24 -p udp—dport 2055 -j ACCEPT
```
9) Save all those changes, so they're persistent:
```
sudo iptables-save
```

Step 4: Set Up the GNS3-GUI

RDP to <GNS3-GUI>

1) Install putty
2) Install the GNS3 GUI software (see Appendix 01 for the full installation steps)
   a. NOTE: the version of the GNS3 GUI must match the server version you installed in step 3.
3) To be able to telnet/ssh to the virtual network devices we're going to build, we need to add a special route.
   a. Open a DOS window (yes, it *IS* called a DOS window. Now get off my lawn!)
   b. Type the following command:
      `route ADD 192.168.122.0 MASK 255.255.255.0 <GNS3-SERVER> -p`

Step 5 Set Up the Orion Engine

Before we get into the nitty gritty of this, I want to take a moment to point out how setting up a lab via a public cloud provider means this step is simply, "go to their store/marketplace/whatever, search for 'SolarWinds,' make a few choices about processors, RAM, and storage, and you're done."

That is, in fact, the benefit of cloud. If someone has created a pre-build template, your work is pretty much done. If this is you, skip to step 3 below.

For the rest of you, let's get to work:

1) RDP to the <Orion Engine>
2) Install the Orion Platform (for detailed information, see Appendix 02)
3) Shut off Windows Firewall for private and public networks

Yes, I'm sure this horrifies some of you. As an alternative, you're welcome to modify the firewall to allow incoming ports for ping, 161, 162, 2055, and so on. But this plan kind of falls apart if you plan to monitor using WMI (which uses randomly chosen ports from 1024 through 65536). In that case, you'd need to open all those ports. So… just shut off Windows Firewall and call it a day.

OK, maybe it's not THAT bad. Since you know all your monitoring is coming from the GNS3 system, and the IP of those devices is using the 192.168.122.0/24 subnet, you can open all those ports from the one specific subnet only, and maintain a little of your security (not to mention dignity among your infosec peers).

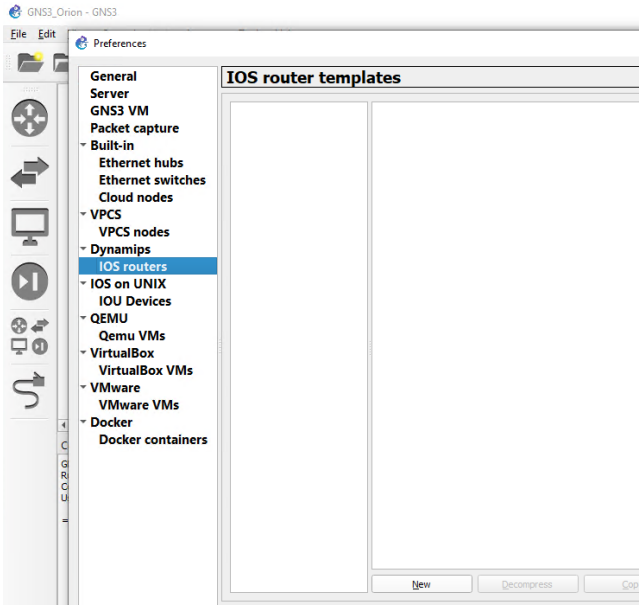
Just as we did with the <GNS3-GUI>, we need to add a special route for telnet and ssh.

4) Open a DOS window (It's STILL called a DOS window. And I told you to get off my lawn!)
5) Type the following command:
   `route ADD 192.168.122.0 MASK 255.255.255.0 <GNS3-SERVER> -p`

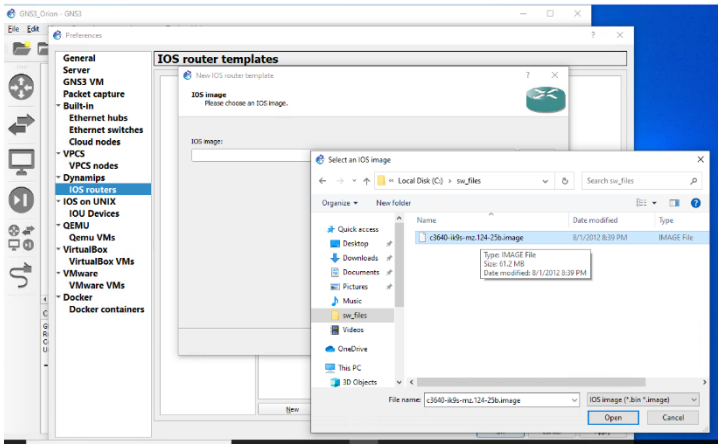
Section 02: Build the Environment

Step 1: Set Up the Virtual Network

1) RDP to the GNS3-GUI
2) Start GNS3
3) If this is the first time GNS3 has started, you'll see the config wizard
    a. Select Remote Server and indicate the IP and port you noted when you set up the GNS3 Server
4) Add an IOS image, so you have a router to work with.
    a. Go to Edit, Preferences
    b. Click "IOS Router" from the sidebar

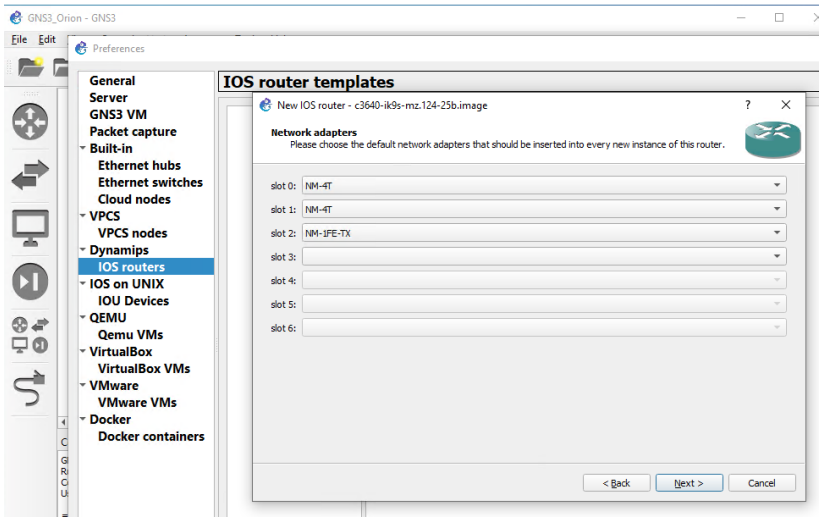


    c. Click "New"
    d. Select "New Image" and browse for the image on your system



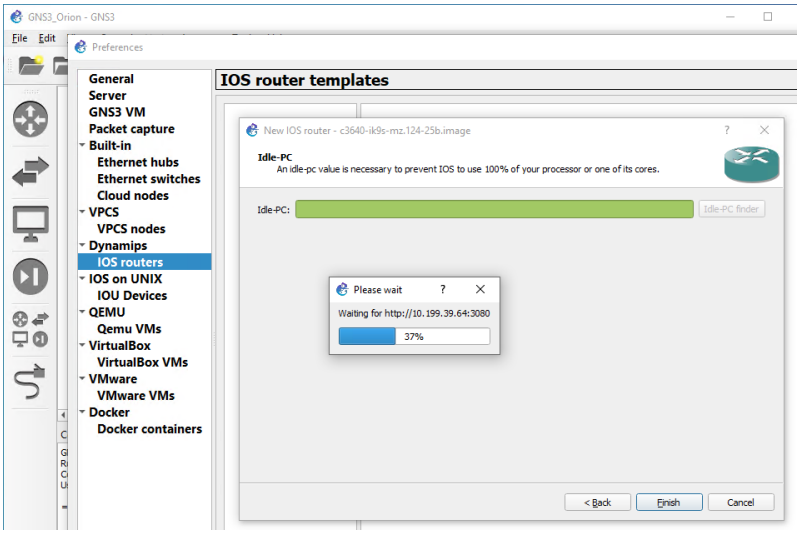
    e. Give the router a name (it can be anything), select the platform, and chassis type
    NOTE: not all devices have a chassis option
    f. On the following screen, confirm RAM

g. Select what goes in each “slot.” This is where you specify the type of network interfaces your virtual router will have—Ethernet, fast Ethernet, gig Ethernet, T1, serial, etc.



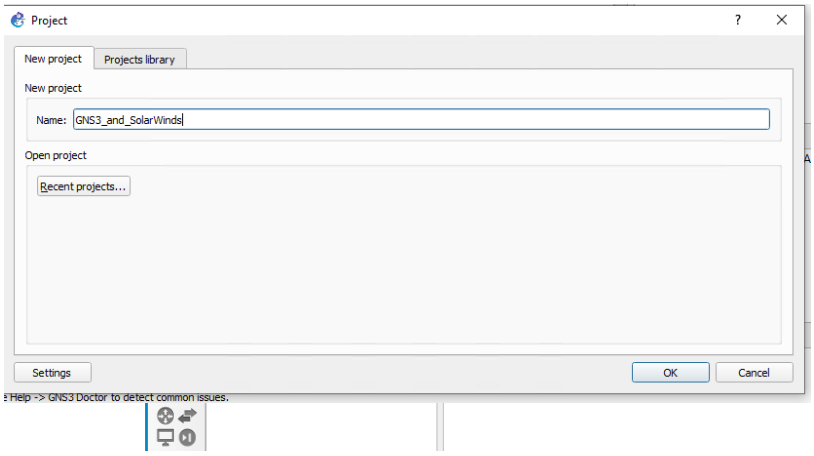
i. For the purpose of Orion monitoring, you MUST make sure you have at least one Ethernet type (regular, fast, or gig) port

h. You’ll also want to have GNS3 validate the Idle-PC setting



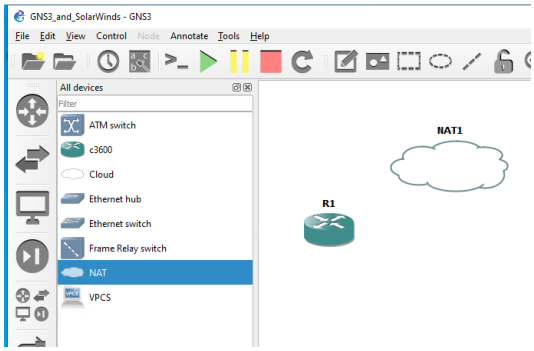
i. Click apply, save, etc. until you’re back at the main screen

5) Back at the main GNS3 screen, create a new project if you haven't already



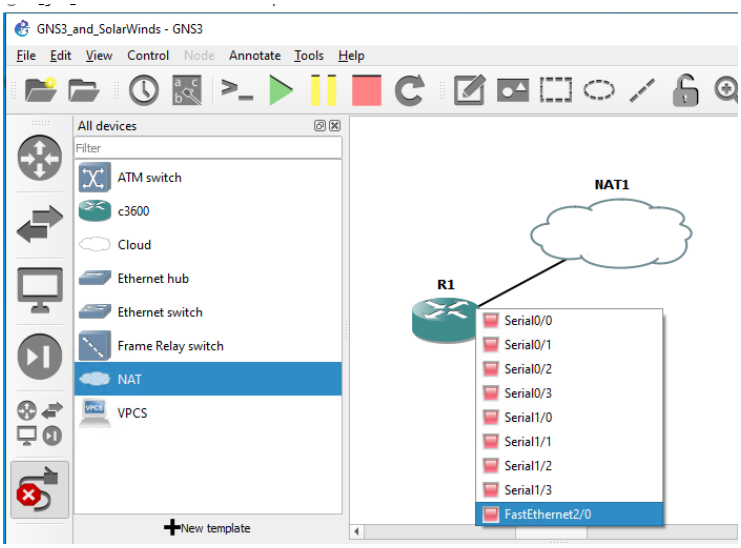
6) Start by adding a router to the design space

7) Add a NAT cloud

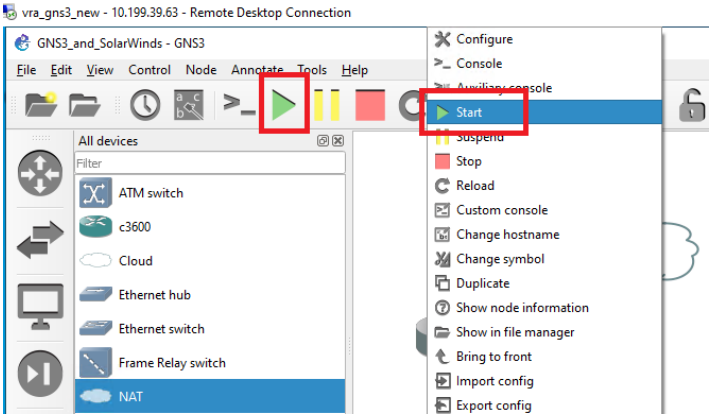


In this part of the guide, we're going with a VERY simple setup—a single router directly connected to the cloud. Appendix 03 has a selection of more complex network configurations and configuration variations you can use if you want to get fancy, or test out things like NetFlow, VoIP, and so on.

8) Connect the Ethernet port of the router to the nat0 port of the cloud



9) Start the router either by clicking the "play" button on the toolbar, or by right-clicking the router and choosing "Start"



Step 2: Configure the Router

Once again, if you are looking for a more complex (or even just realistic) network design, check out Appendix 03.

1) Right-click the router and choose “console”
2) Hit enter a couple times to get to an “R1” prompt
3) Get into configuration mode
   conf t
4) The following set of commands will set up the username and password for local and remote logins as well as setting up ssh instead of the default telnet connection type

NOTE: The router will now support ssh.

- The login username will be “gns3” and the password will be “solarwinds”
- The password when you go into “enable” mode (i.e., type the `en` command) is “solarwinds”
- The “domain” of the router is “solarwinds.gns3”
- The SNMP read-only string is “solarwinds”
- The SNMP read-write string is “solarwinds”

```
conf t
username gns3 password solarwinds
enable secret solarwinds
service password-encryption
ip domain-name solarwinds.gns3
snmp-server community solarwinds ro
snmp-server community solarwinds rw
crypto key generate rsa
1024
line vty 0 4
transport input ssh
login local
password 10 solarwinds
exit
line console 0
logging synchronous
login local
exit
```

5) Configure the interface that will connect to the monitoring system

```
int fa2/0
ip addr dhcp
no shut
end
wr mem
```

Make sure you see a DHCP message indicating the interface has been assigned an IP address. It will look like this:

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int fa2/0
R1(config-if)#ip addr dhcp
R1(config-if)#no shut
R1(config-if)#
*Mar  1 00:00:42.703: %LINK-3-UPDOWN: Interface FastEthernet2/0, changed state to up
*Mar  1 00:00:43.703: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet2/0, changed state to up
R1(config-if)#
*Mar  1 00:00:57.495: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet2/0 assigned DHCP address 192.168.122.115, mask 255.255.
255.0, hostname R1

R1(config-if)#
```

Make a note of this IP, since you'll want to use it for both testing and troubleshooting, and more importantly for the actual monitoring we'll do later.

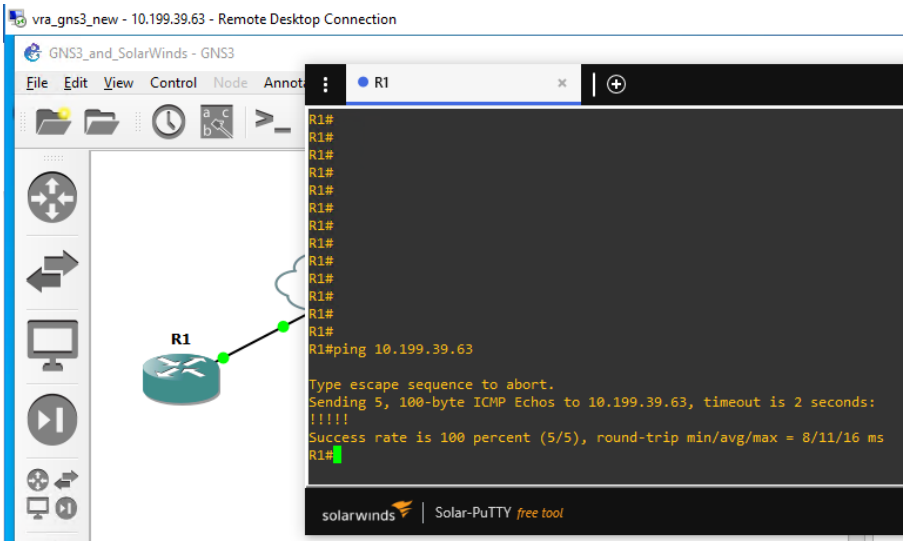
At this point, you have a "network" ready for monitoring by the SolarWinds Orion Platform. It's not particularly awe-inspiring, but it'll get the job done and proves the concept.

Section 03: Let's Do Some Testing

At this point, in theory at least, everything should be set up and running. Then again, we all know "theoretically" is Murphy's Greek cousin, so let's validate those assumptions.

Test 1: Connectivity From the Virtual Network Device(s)

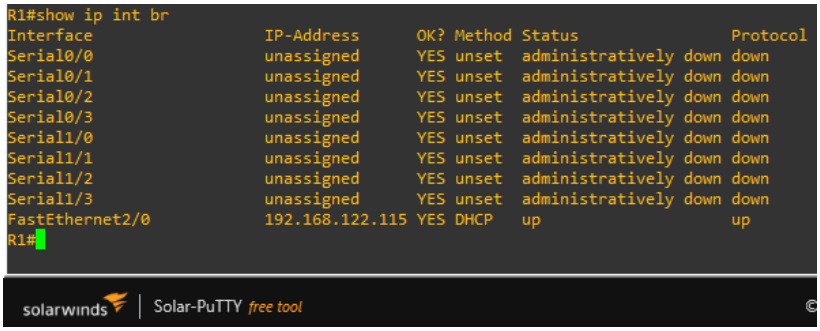
1) RDP to the <GNS3-GUI> system
2) If GNS3 isn't running, start it up
3) If the router we set up isn't running, start it
4) Ping from the router to the <GNS3-GUI>
    a. Right-click the router and choose "console"
    b. Hit enter a couple of times
    c. Enter the username and password when prompted
    d. Type "ping" and the GNS3-GUI machine's IP



5) Ping from the router to the <Orion-engine>
    a. You should already have a console open
    b. Type "ping" and the Orion engine's IP

Test 2: Connectivity From the <GNS3-GUI> Machine

1) You should still be RDPed to the GNS-GUI system from the previous step
2) Open the DOS prompt (don't start with me, whippersnapper)
3) Ping the virtual router's Ethernet interface, which should be at a 192.168.122.x address
    a. If you aren't sure of the IP, go back to the console
    b. Type "`show ip interface brief`" and hit enter



```
R1#show ip int br
Interface              IP-Address      OK? Method Status                Protocol
Serial0/0              unassigned      YES unset  administratively down down
Serial0/1              unassigned      YES unset  administratively down down
Serial0/2              unassigned      YES unset  administratively down down
Serial0/3              unassigned      YES unset  administratively down down
Serial1/0              unassigned      YES unset  administratively down down
Serial1/1              unassigned      YES unset  administratively down down
Serial1/2              unassigned      YES unset  administratively down down
Serial1/3              unassigned      YES unset  administratively down down
FastEthernet2/0        192.168.122.115 YES DHCP   up                    up
R1#
```

solarwinds | Solar-PuTTY *free tool* ©

4) (Still from the DOS window on the GNS-GUI machine) Ping the Orion-engine IP

Test 3: Connectivity From the <Orion-Engine>

1) RDP to the Orion-engine
2) Open a DOS window (I'm not gonna say it again)
3) Ping the GNS3-GUI IP
4) Ping the virtual router's Ethernet interface, which should be at a 192.168.122.x address

If you've gotten through all these tests, we're ready to start monitoring!

Section 04: It's Time to Monitor!

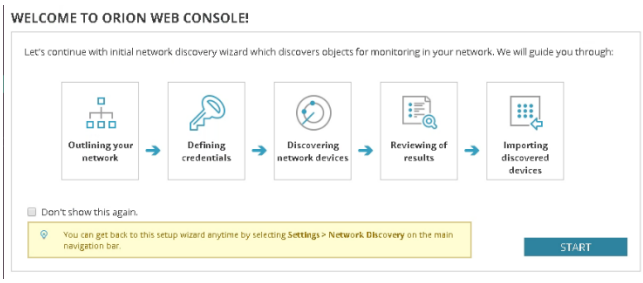
If all your testing was successful, there's no reason to think setting up monitoring isn't going to be a breeze.

<ahem>

OK, I'll admit it: if you've worked in IT for more than 15 minutes, you know there are PLENTY of reasons to think this might go south. But even so, we here at SolarWinds are adventurous problem solvers and we urge you to embrace this spirit. Let's dive in and hope for the best.

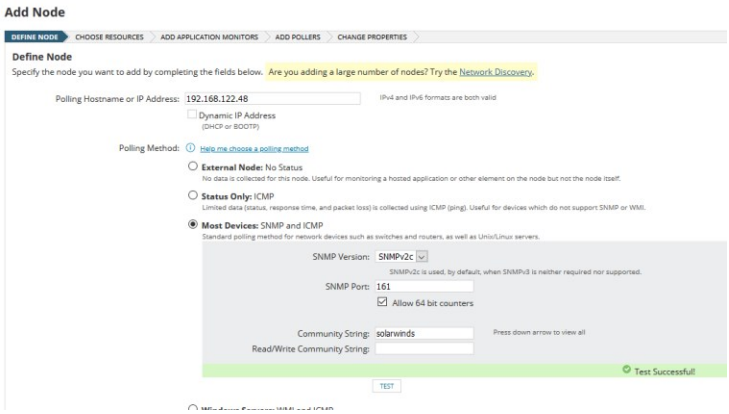
Step 1: Start up the Orion Platform for the first time.

1) Open your web browser and go to http://<orion-engine> (that's the IP you noted earlier).
   That's right, no RDP necessary. What do you think this is, a Perl script running on a Windows NT box in 1997?*
2) If you are logging in for the first time, you probably haven't set the Admin username. Do it now. We'll wait.
3) Next, you'll get the first-time login splash screen.



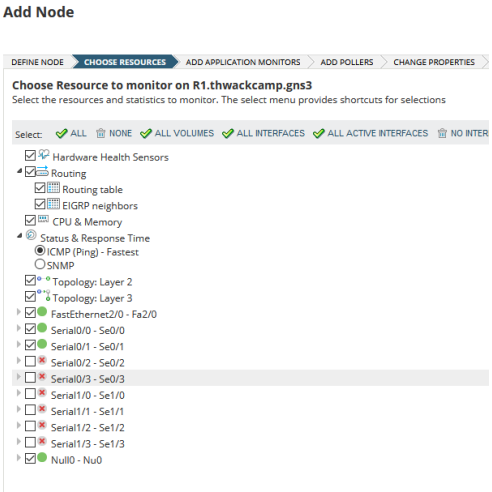
While network discovery is awesome and usually exactly the right way to start your Orion Platform experience, in this case our network is very small, and I want to highlight a few aspects of adding virtual GNS3 devices.

4) Cancel out of this splash screen and you'll end up at the home screen.
5) Go to Settings, All Settings, and click "Add Node."
6) On the add node page, enter the IP of the router you set up in GNS3.
   Put in the SNMP read-only string (it's "solarwinds" if you're following our example) and click the "test" button.



* Not that I ever did anything like that. Not me. Nope.

7) You can leave the rest of the screen alone for now and click “Next.”
8) On the next screen, you’ll see resources—CPU, RAM, interfaces, and so on. Select the ones you want to monitor and click “Next.”



9) The next screen will ask you about Universal Device Pollers (UnDP). We’ll skip this step for now and just click “Next.”
10) The final screen confirms the primary settings you’d set up initially and allows you to add custom properties. Those are project for another day, so you can click “OK, ADD NODE” and be done.

At this point, you’re back at the main Orion Platform screen and guess what? You’re monitoring a GNS3 device! Give yourself a high five.

The Mostly Un-Necessary Summary

Obviously, this is only scratching the surface of what you can do. We're going to keep developing this guide in the coming weeks, months (and yeah, probably years too, seeing as this all started back in 2014) to continue to show you how flexible and valuable this type of setup can be.

Enjoy enhancing and exploring this new option for creating reliable test environments for both your network and your monitoring solutions.

I also encourage you to share your experiences on the THWACK and GNS3 communities.

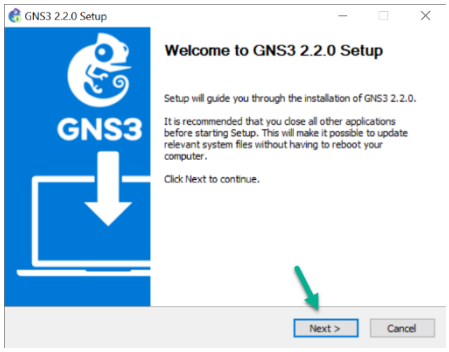
Appendix 01: GNS3 GUI Installation Detailed Instructions

For the latest updates to this process, check out the online installation document:
https://docs.gns3.com/docs/getting-started/installation/windows/

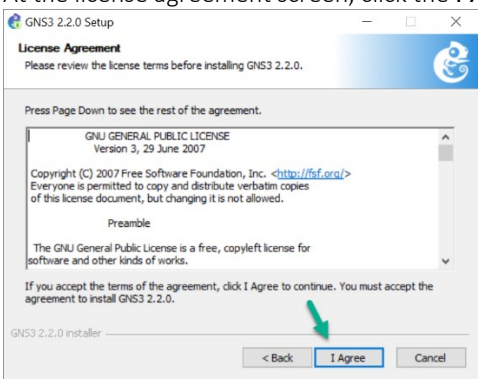
Locate the GNS3 installer and double-click it to start the installation. If displayed, click the **Run** button to start the GNS3 installation (it's not possible to take a screenshot of the UAC prompt for this, but allow setup to continue).

NOTE: The GNS install process will run multiple installation wizards. If the process seems to have stalled, check if additional windows have been opened and are waiting for your input.

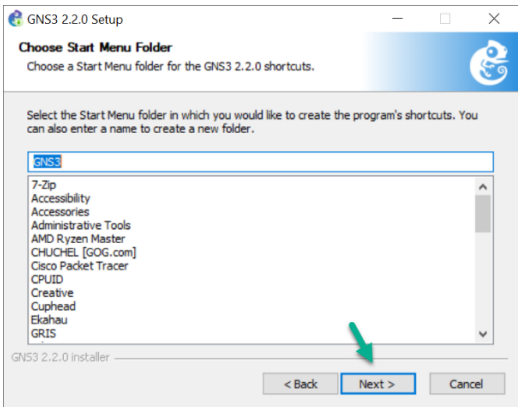
The GNS3 Setup wizard displays. Click **Next >** to start the installation:

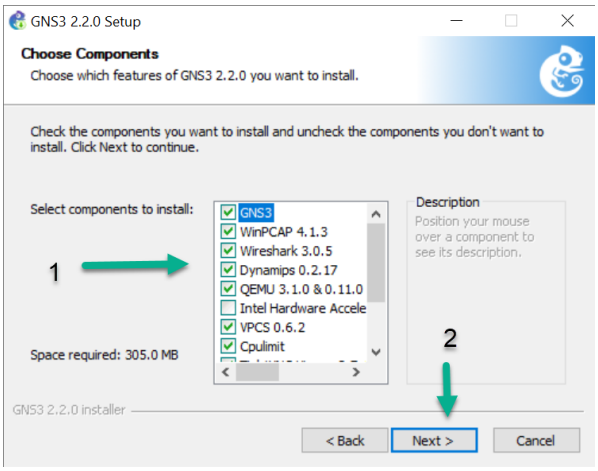


At the license agreement screen, click the **I Agree** button to continue the installation:



Select the Start Menu folder for the GNS3 shortcut. The default is the GNS3 folder. Click **Next >** to continue the installation:





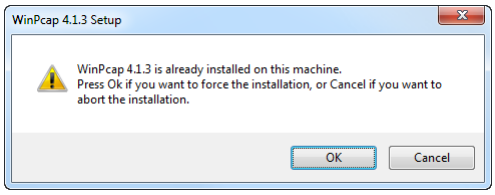
GNS3 comes bundled with various prerequisite and optional software. By default, most software is selected for installation, but you can decide to only install specific software. If you're unsure, leave all software selections at their default selection and click Next > to continue the installation:

Here's a brief description of software:

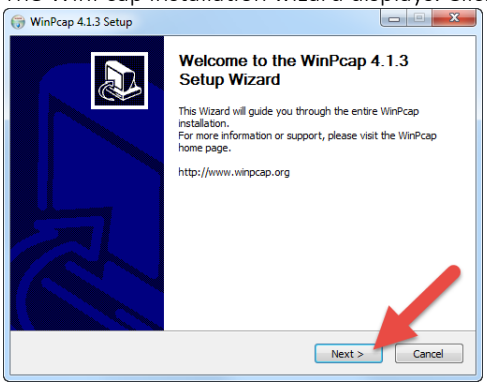
Item	Required	Description
WinPCAP	Required	Required to connect GNS3 to your computer network. Used by cloud and NAT nodes to allow your projects to communicate with the outside world.
Npcap	Optional	Modern replacement to WinPCAP to fix issues with Win10 but is less tested than WinPCAP. Install Npcap with the "WinPcap API-compatible Mode" option selected, if using without WinPcap. Npcap can co-exist with WinPcap, if this option is **not** selected.
Wireshark	Recommended	Allows you to capture and view network traffic sent between nodes.
Dynamips	Required	Required to run a local installation of GNS3 with Cisco routers. Only unselect if you're going to exclusively use the GNS3 VM.
QEMU 3.1.0 and 0.11.0	Optional	A computer emulator used to emulate a full computer, which could for example be Linux. The older Qemu version 0.11.0 is installed to support old ASA devices. It is recommended to use the GNS3 VM instead.
VPCS	Recommended	A very light PC emulator supporting basic commands like ping and traceroute.
Cpulimit	Optional	Used to avoid QEMU using 100% of your CPU (when it's running) in some cases like with the old ASA devices.
GNS3	Required	The core GNS3 software. This is always required.
TightVNC Viewer	Recommended	A VNC client used to connect to appliance graphical user interfaces.
Solar-Putty	Recommended	The new default Console application.
Virt-viewer	Recommended	Alternate displayer of Qemu desktop VMs with qemu-spice pre-installed.
Intel Hardware Acceleration Manager (HAXM)	Optional	Only available on systems with Intel CPUs (and VT-X enabled), that are **not** using Hyper-V. Used for hardware acceleration of Android emulation, as well as QEMU.

Choose an install location. The default location is C:\Program Files\GNS3. Then click Install:
The output displayed will depend on what you have selected to install.

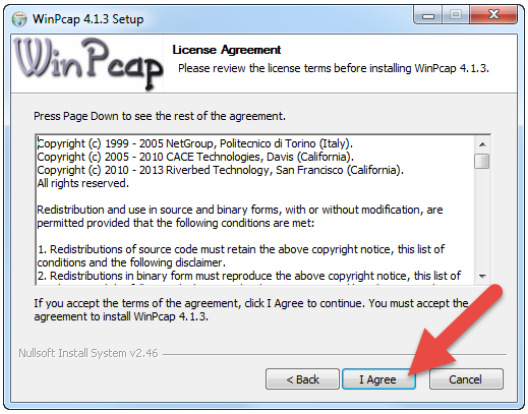
If WinPcap is already installed, a warning message is displayed. It's not necessary to reinstall WinPcap. Click Cancel if you don't want to reinstall WinPcap and go directly to the next step. Click OK to continue re-installing WinPcap:



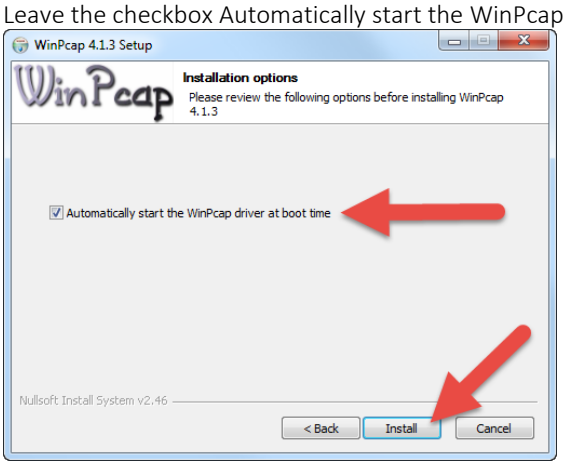
The WinPcap installation wizard displays. Click Next > to continue the installation:



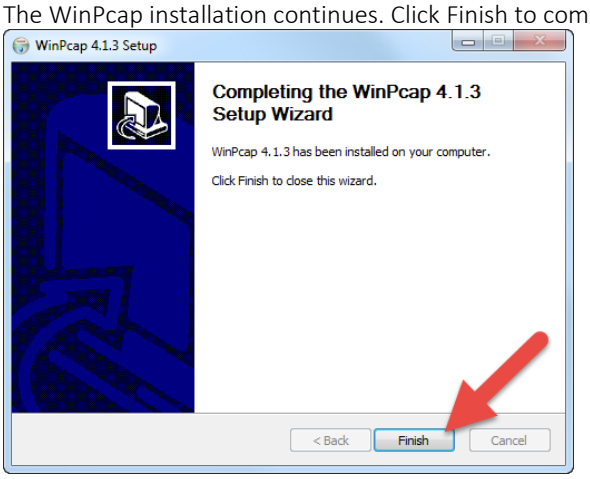
The WinPcap License Agreement displays. Read the agreement and if you agree, click the I Agree button to continue with the installation:



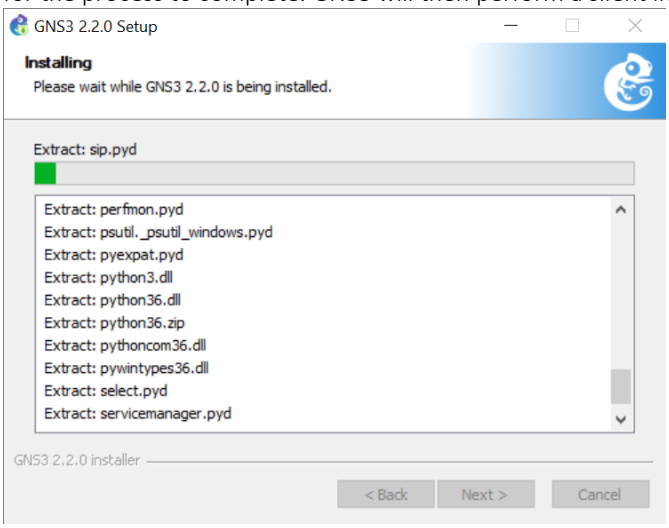
Leave the checkbox Automatically start the WinPcap driver at boot time checked and click Install:



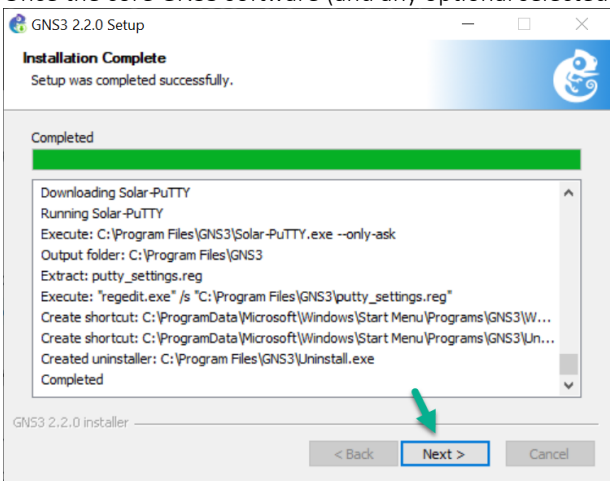
The WinPcap installation continues. Click Finish to complete the installation:



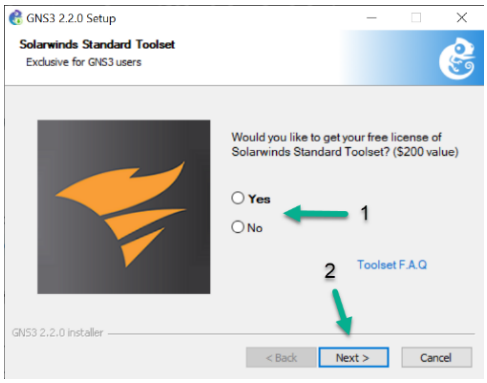
If you selected for Wireshark to be installed, the GNS3 setup software will download the Wireshark install files. Wait for the process to complete. GNS3 will then perform a silent install of Wireshark. Wait for the process to complete:



Once the core GNS3 software (and any optional selected items) is installed, Click Next>

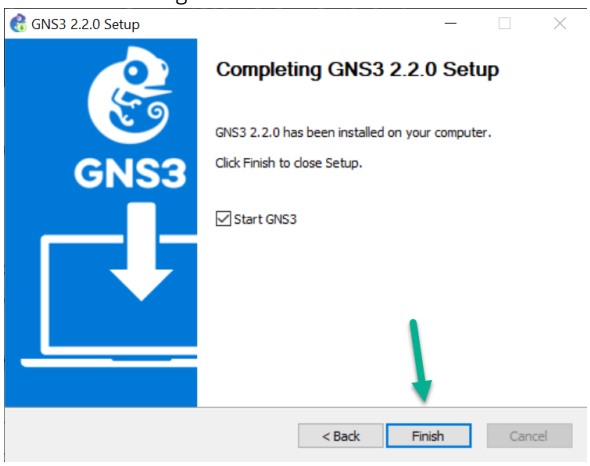


Optional: If you're interested, install the SolarWinds Standard Toolset. This is a free evaluation (a $200 value). Otherwise Select No and click Next > to continue:



If you elect to download the Standard Toolset, you'll be redirected to a webpage on SolarWinds, where you enter registration information. After registration and clicking the Download Free Toolkit link, you'll be redirected to an informational page asking you to check your email, as well as instructions on how to activate your license for the Standard Toolset.

Congratulations! You have successfully installed GNS3. A browser window will open showing additional help and options. Leave the Start GNS3 checkbox enabled and click Finish to complete the GNS3 installation. Click here for a walkthrough of using the Setup Wizard to configure and use the Local Server.



Appendix 02: Orion Platform Installation Detailed Instructions

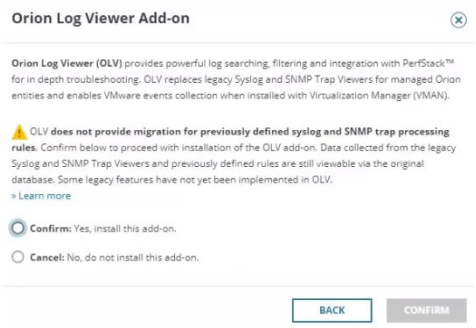
For the latest version of these installation instructions, check out the online guide:
https://documentation.solarwinds.com/en/Success_Center/orionplatform/Content/Install-OrionProducts.htm

NOTE 1: This installation presumes you already have Windows Server 2016 (or higher) installed, patched, and ready to go for the Polling Engine, and you have access to a server running Microsoft SQL Server 2016 (or higher) to install the database.

NOTE 2: Some third-party software, such as .NET 4.8, is required. If it's not found on the server, it's downloaded and installed when you run the installer.

1) Run the installer .exe file on your primary Orion server as LOCAL Administrator. The Select Products page lists the products you can install and any existing products to be upgraded.
2) Select the products to install (for this guide, we're suggesting AT LEAST Network Performance Monitor (NPM), although you can choose additional modules if you want) and click Next.
3) If the Add-ons page is displayed:

- Click More Info to learn more about the available component.
- Select Confirm to install the component or Cancel if you don't want to install it.
- Click Confirm to close the dialog box, and then click Next.



The System Check page asks you to confirm you backed up your database.

4) Under System Check Confirmations, click Choose how to proceed. If you have backed up your database, select Confirm and then click Confirm.

   If not, you should back it up now. New products and versions can modify your database tables. Click here for information about SolarWinds Backup.
5) On the Install Report page, review the information under System Check Results.

   The installer runs a series of checks per product to verify your server meets system requirements and recommendations. If your environment does not meet specifications, the installer displays one or more messages:

- Informational and warning messages recommend actions and best practices to optimize performance. These do not block the installation.
- Critical issues describe changes required to support the products. These block the installation until they're resolved.
- Investigate and resolve any issues:
  a. Click the details link to display additional information and suggested resolutions.
  b. To save the list of issues, click Save System Check Results.
  c. After resolving any blocking issues, click Run Checks Again.
- Click Next.

6) Review the EULA. If you agree, click I accept, and then click Next to begin the installation.

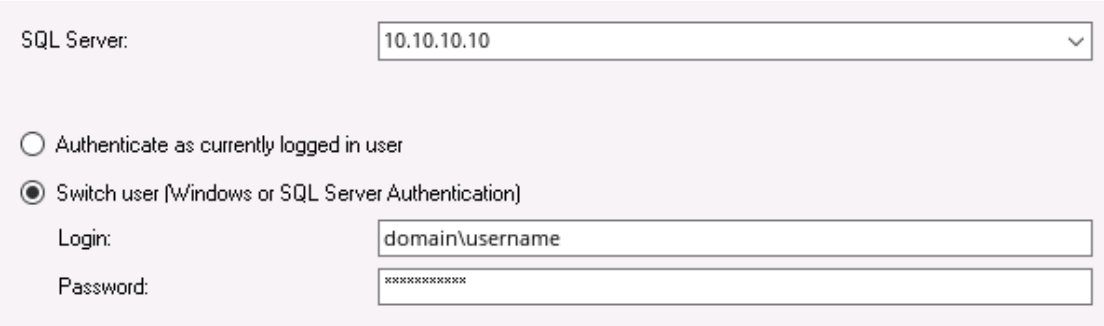
The Installation page displays progress messages. If the installer encounters any issues, the installation stops, so you can resolve them. The installer might run multiple product installations before running the Configuration wizard.

If a reboot is required as part of the installation, a message is displayed.

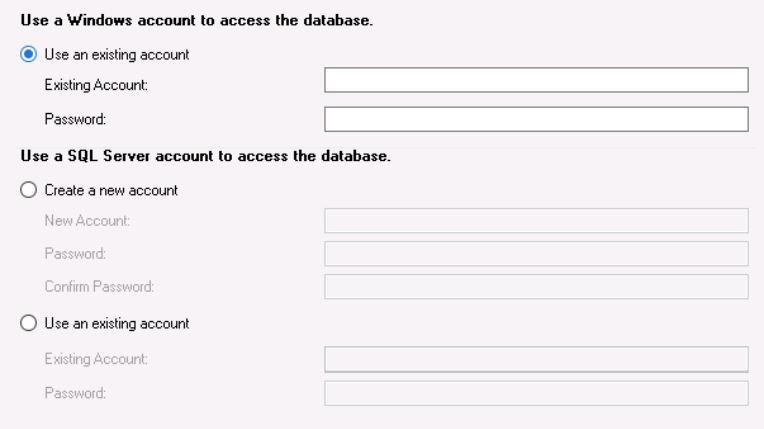
If database configuration is required, the Configuration wizard automatically opens. Depending on your products, the wizard might include options and pages not described here.

7) On the Welcome page, click Next.
8) If prompted to stop services, click Yes.
9) If you performed a Standard installation with an existing SQL Server database, select one of the following for authentication:
10) Authenticate as currently logged in user: Pass through authentication to the database server using the account currently logged in for installing the Orion Platform product.
11) Switch user: Provide credentials automatically detected as either SQL or Windows credentials, allowing Windows authentication for the initial setup even if the Orion server is not joined to a domain or the current account does not have permissions to the database server.

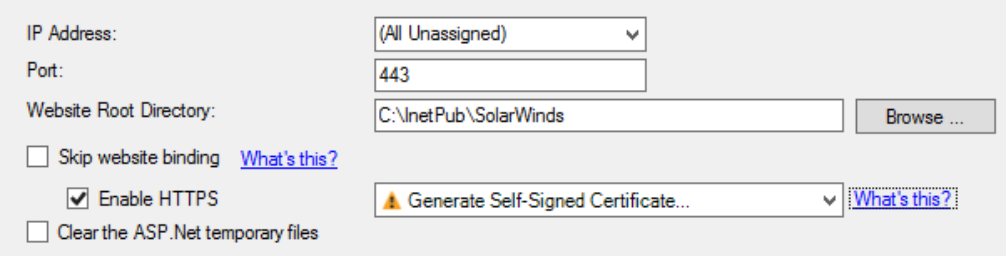
If you intend to use Windows authentication for the Orion Platform, remember to exempt that user account from any password change policies. An expired password will cause the Orion Platform to stop data collection and interrupt Orion Web Console access.



12) On the Database Settings page, select your existing Orion database, or create a new database for a new installation, and click Next.
13) On the Database Account page, create an account or specify an existing account the polling engine and Orion Web Console will use to access the database. The account can be a Windows or SQL Server account.



14) On the Website Settings page:
15) Select All Unassigned unless your environment requires a specific IP address for the Orion website. If SSL is selected, port 443 is used. Otherwise, port 80 is used.
16) Specify the Port and the Website Root Directory where the system installs the web console files. If you specify any port other than 80, include the port in the URL used to access the Orion Web Console.
17) To configure SSL, click Enable HTTPS and select your SSL certificate.
    If a certificate isn't available, select the option to Generate Self-Signed Certificate. The Configuration wizard automatically generates a self-signed certificate issued to the hostname or FQDN and adds it to the trusted certificate store.



If you select Skip website binding, the Configuration wizard doesn't make changes within the website configuration in your IIS. This option prevents IP address, port, and SSL certificate options.

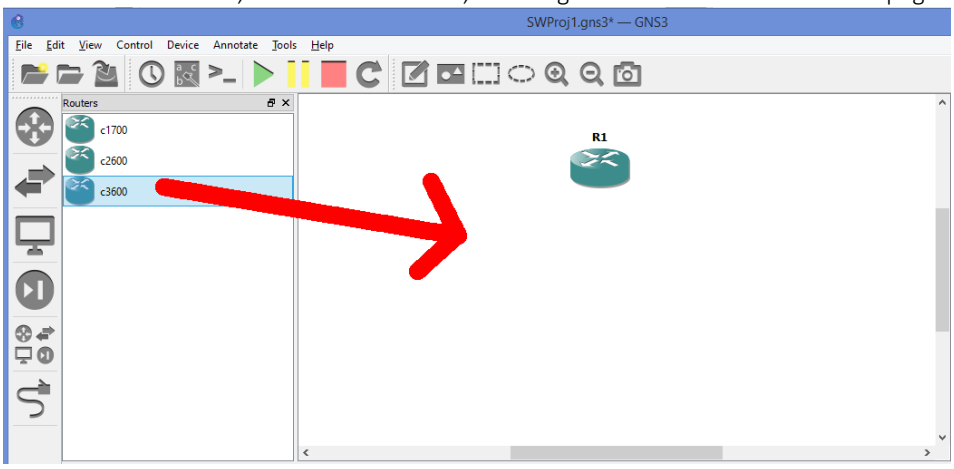
18) If prompted to create a directory or website, click Yes.
19) Review the list of services to install and click Next.
20) Click Yes if prompted to disable the SNMP Trap Service and enable the SolarWinds Trap Service.
21) On the Completing the Orion Configuration Wizard page, click Next.
22) When the configuration is complete, click Finish to launch the Orion Web Console.
    If the Orion Web Console doesn't open automatically (for example, if it times out before opening), do one of the following to open it manually:
23) Click Start > All Programs > SolarWinds > Orion Web Console.
24) Open a web browser on your Orion server and enter http://ipAddress or http://hostname, where ipAddress is the IP address of your server and hostname is the host name of your server. Enter https:// if you selected SSL.

Appendix 03: Sample Network Configurations

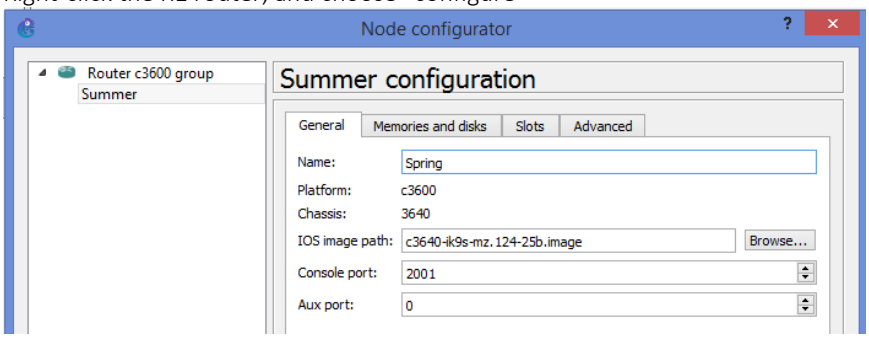
Example 1: Single Router Connected to NAT Cloud

This example sets up one router with a FastEthernet interface connected to the NAT cloud, which provides a path out to other real resources.

1) From the main screen, click the router icon, and drag a 3600 router onto the main page:

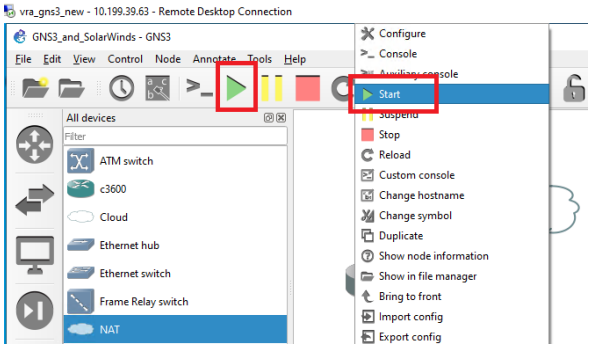


2) Right-click the R1 router, and choose "configure"

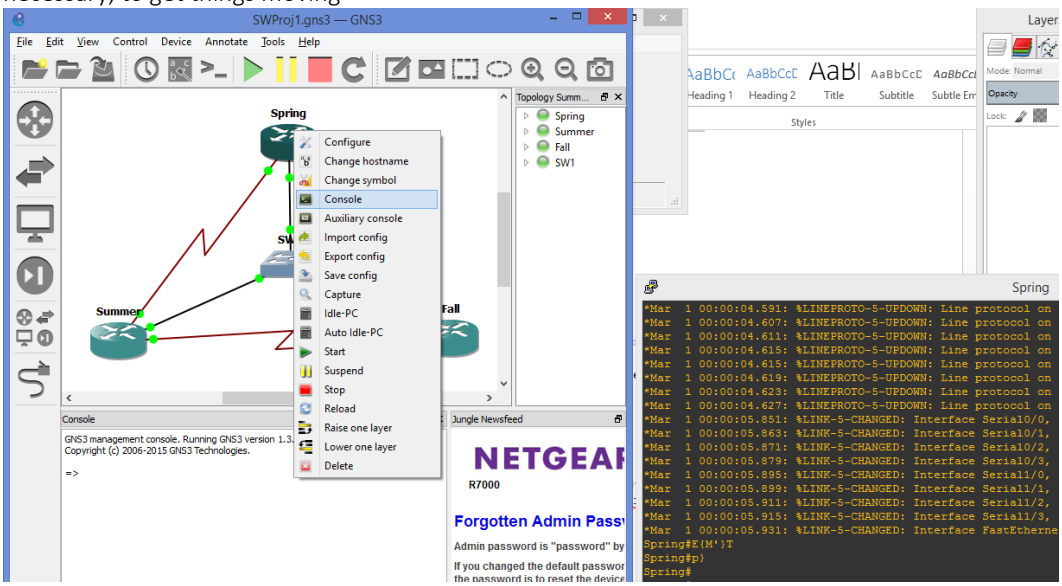


   a. On the General tab, set the name to "Spring"
   b. On the Slots tab, make sure slot 0 to NM-1FE-TX (FastEthernet)

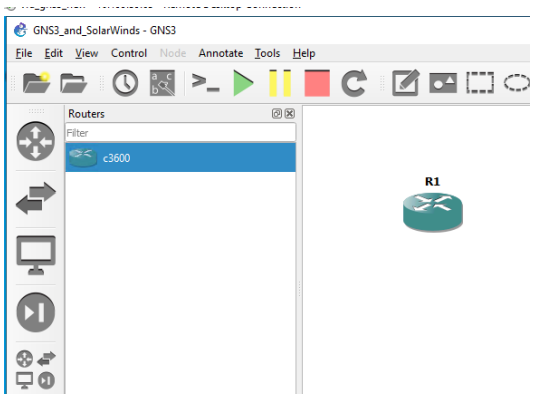
3) Click OK to finish
4) Click the "play" button to start your router



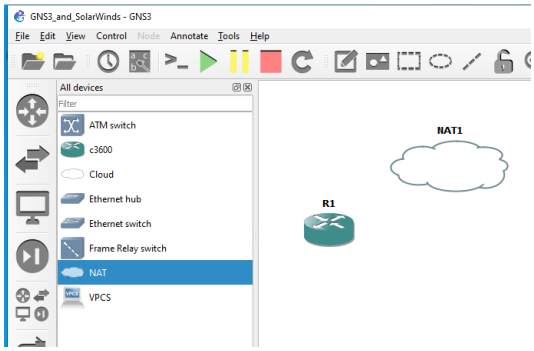
5) Right-click “Spring” and choose “Console “to open a telnet/ssh terminal. Hit ENTER a couple of times if necessary, to get things moving



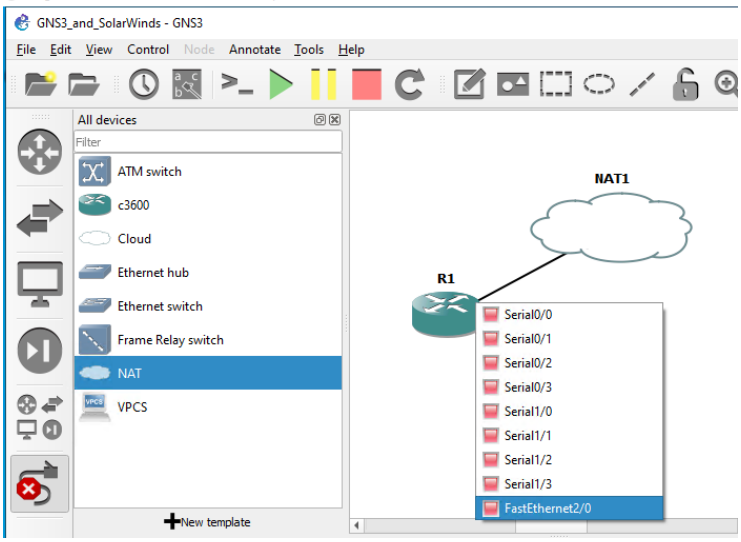
6) Set up SNMP
   a. Type this to enter config mode
      `configure terminal`
   b. Type this to set up a SNMP read-only string, so you can monitor with SNMP
      `snmp-server community GNS3plusSWrocks ro`
7) Set up the fa2/0 interface
   c. Type these commands to set up the third (FastEthernet) interface
      ```
      int Fa2/0
      ip address dhcp
      no shutdown
      end
      write memory
      ```
8) Start by adding a router to the design space



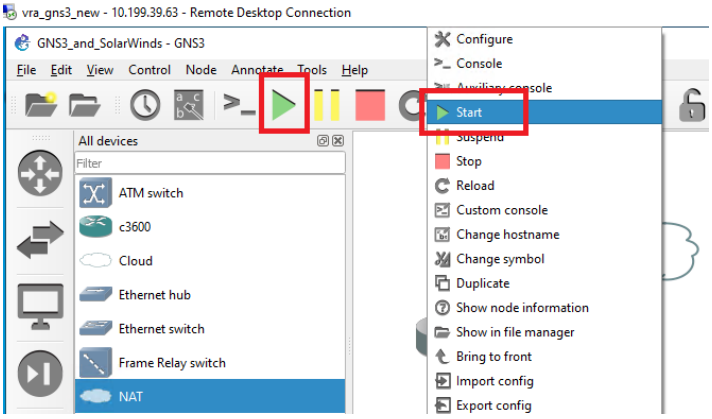
9) Add a NAT cloud



10) Connect the Ethernet port of the router to the NAT0 port of the cloud

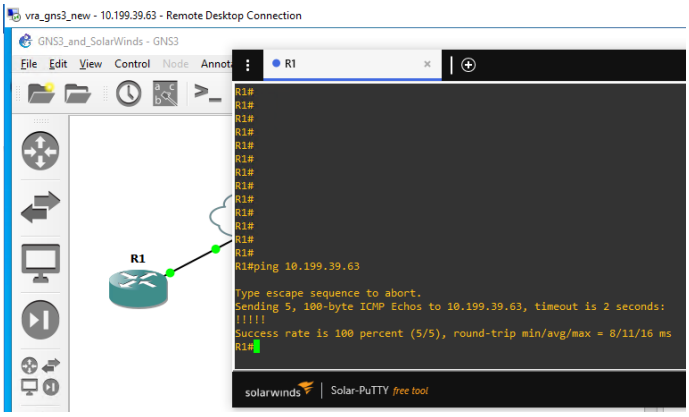


11) Start the router either by clicking the "play" button on the toolbar, or by right-clicking the router and choosing "Start"



12) TESTING PHASE

Open the console on the router and make sure you see the DHCP message for fa2/0

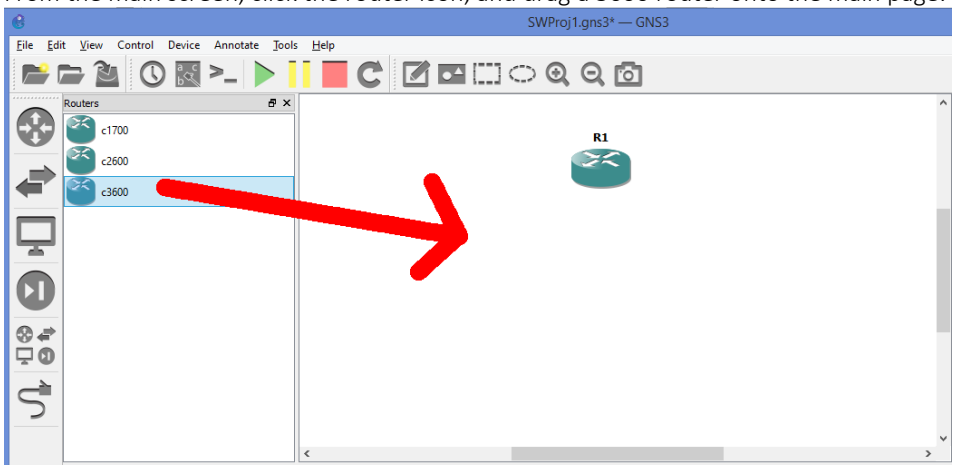


Example 2: Three Routers Connected to NAT Cloud

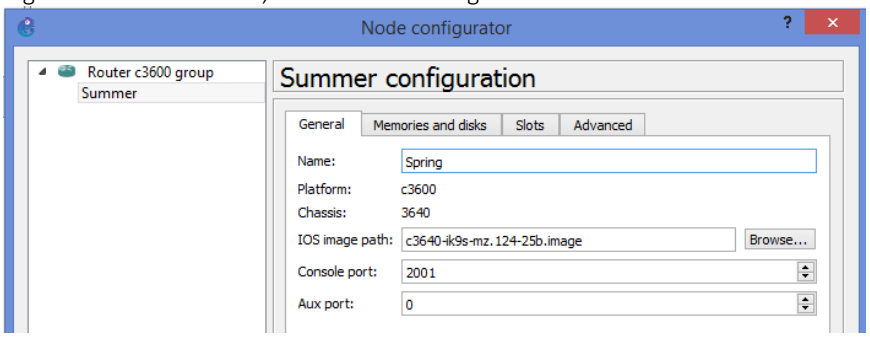
Overview

This example sets up three routers communicating to each other on T1 circuits using the EIGRP routing protocol. Each router also connects via a FastEthernet interface to a common switch, which is connected to the NAT cloud and provides the path out to other real resources.

1) From the main screen, click the router icon, and drag a 3600 router onto the main page:



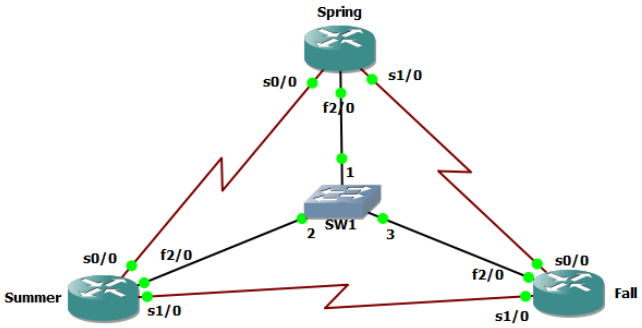
2) Right-click the R1 router, and choose "configure"



   a. On the General tab, set the name to "Spring"
   b. On the Slots tab, make sure slots 0 and 1 are set to NM-4T (serial)
   c. Set slot 2 to NM-1FE-TX (FastEthernet)

3) Click OK to finish, and repeat these steps for Router 2 and 3
   Naming them Summer and Fall, respectively

With the routers all placed and provisioned, we need to configure them.
First, a picture:



And here's a description of the interfaces on each device

Router1 Name: Spring

- Interface s0/0 (serial):
  - 10.1.1.1/24 (i.e.: gateway 255.255.255.0)
  - Connects to interface s0/0 on Summer
- Interface s0/1 (serial):
  - 10.1.3.2/24
  - Connects to interface s0/0 on Fall
- Interface f2/0 (FastEthernet):
  - Connects e1 of Switch 1
  - Will obtain a 192.1168.122.0/24 IP address via DHCP

Router2 Name: Summer

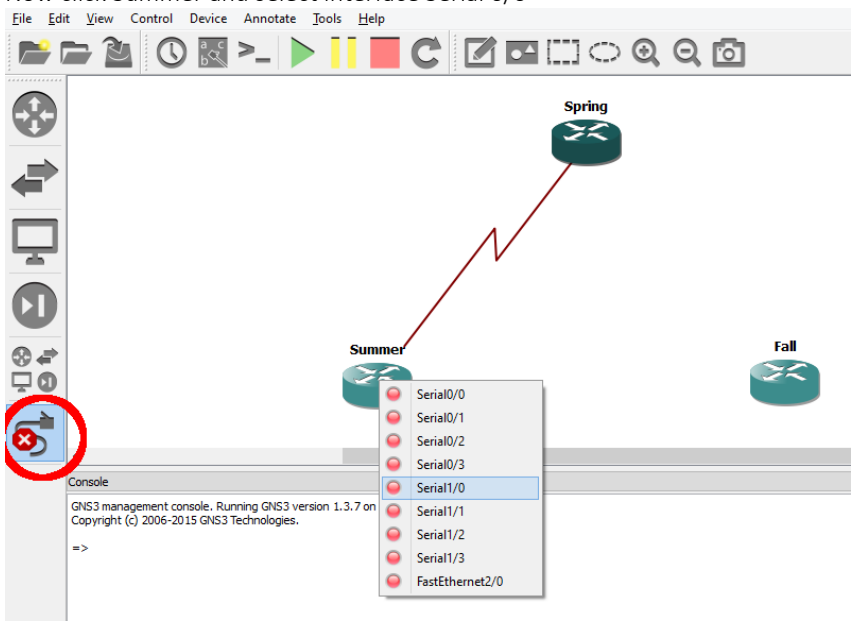
- Interface s0/0:
  - 10.1.1.2/24
  - Connects to s0/0 on Spring
- Interface s0/1:
  - 10.1.2.1/24
  - Connects to s0/1 on Fall
- Interface f2/0 (FastEthernet):
  - Connects e2 of Switch 1
  - Will obtain a 192.1168.122.0/24 IP address via DHCP

Router3 Name: Fall

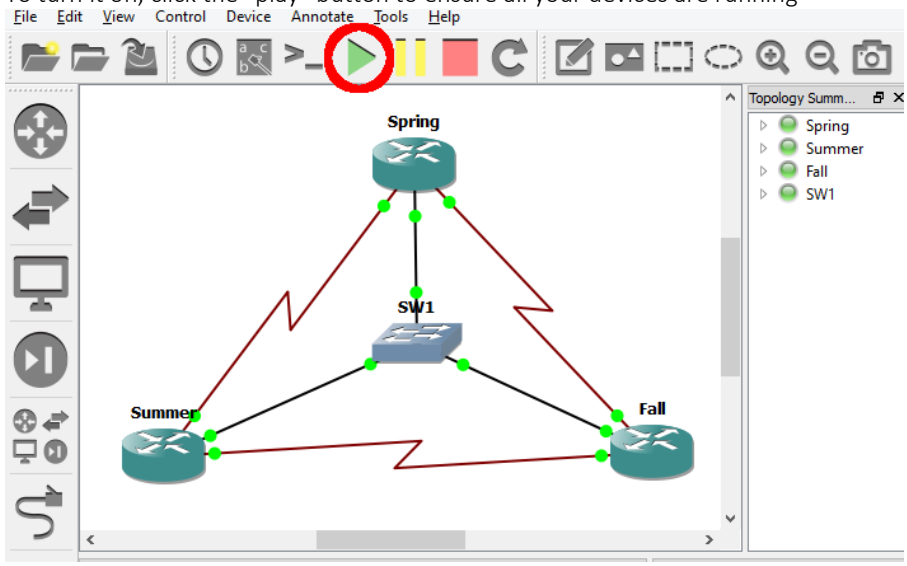
- Interface s0/0:
  - 10.1.2.2/24
  - Connects to s0/1 on Summer
- Interface s0/1:
  - 10.1.2.2/24
  - Connects to s0/1 on Spring
- Interface f2/0 (FastEthernet):
  - Connects e3 of Switch 1
  - Will obtain a 192.1168.122.0/24 IP address via DHCP

With this design in mind, let's get to configuring!

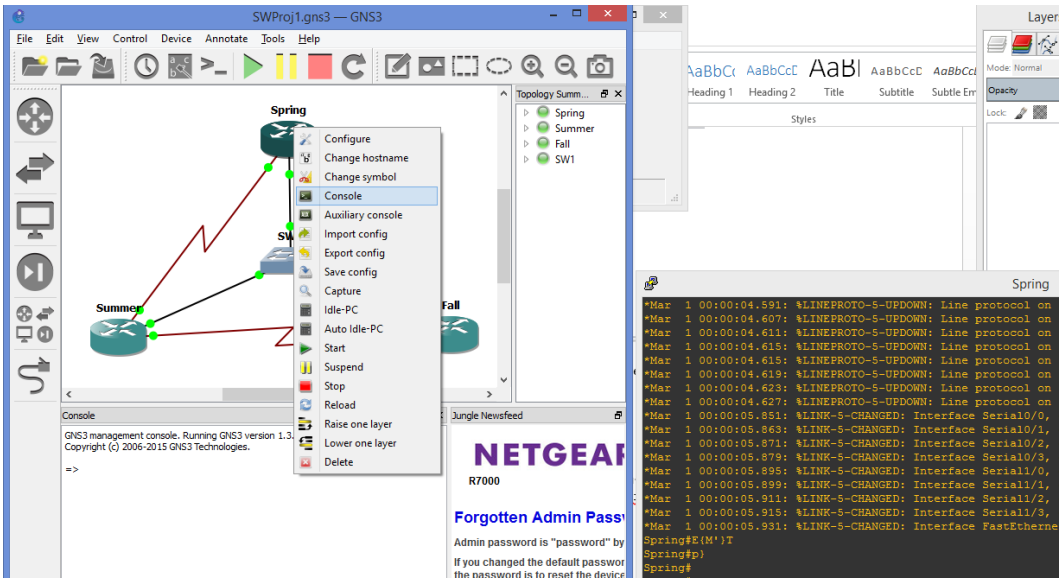
1) Click the "connection" button
2) Click Spring and select interface Serial0/0
3) Now click Summer and select interface Serial 0/0



4) Continue to connect routers as follows:
    d. Spring, Serial1/0 to Fall, Serial1/0
    e. Fall, Serial0/0 to Fall, Serial1/0
5) Click the switch icon and drag an "Ethernet switch" onto the screen.
6) Click the connector icon again and connect each router's FastEthernet2/0 interface to the switch
7) To turn it on, click the "play" button to ensure all your devices are running



8) Right-click "Spring" and choose "Console "to open a telnet/ssh terminal. Hit ENTER a couple of times if necessary, to get things moving



9) Set up SNMP
   f. Type this to enter config mode
      `configure terminal`
   g. Type this to set up a SNMP read-only string, so you can monitor with SNMP
      `snmp-server community GNS3plusSWrocks ro`
10) Set up the interfaces
   h. Type this to edit the first interface:
      `interface s0/0`
   i. Type these commands to set up the first interface
      `ip address 10.1.1.1 255.255.255.0`
      `no shutdown`
   j. Exit back to interface mode
      `exit`
   k. Type these commands to set up the second interface
      `interface s1/0`
      `ip address 10.1.3.2 255.255.255.0`
      `no shutdown`
   l. Exit back to interface mode
      `exit`
   m. Type these commands to set up the third (FastEthernet) interface
      `int Fa2/0`
      `ip address dhcp`
      `no shutdown`
   n. Exit back to interface mode and set up EIGRP routing
      `exit`
      `router EIGRP 1`
      `network 10.1.1.0 0.0.0.255`
      `network 10.1.3.0 0.0.0.255`
      Exit all the way out and save your config
      `end`
      `write memory`

11) Set up EIGRP on the other two routers

o. Right-click on the router "Spring" and open a console, and then enter the following commands:

```
configure terminal
snmp-server community GNS3plusSWrocks ro
interface s0/0
ip address 10.1.1.2 255.255.255.0
no shutdown
exit
int s1/0
ip address 10.1.2.1 255.255.255.0
no shutdown
exit
int Fa2/0
ip address dhcp
no shutdown
exit
router EIGRP 1
network 10.1.1.0 0.0.0.255
network 10.1.2.0 0.0.0.255
end
write memory
```

p. Right-click on the router "Fall" and open a console, then enter the following commands:

```
configure terminal
snmp-server community GNS3plusSWrocks ro
interface s0/0
ip address 10.1.3.1 255.255.255.0
no shutdown
exit
int s1/0
ip address 10.1.2.2 255.255.255.0
no shutdown
exit
int Fa2/0
ip address dhcp
no shutdown
exit
router EIGRP 1
network 10.1.3.0 0.0.0.255
network 10.1.2.0 0.0.0.255
end
write memory
```

12) Check to ensure your network is working by issuing the following commands on all three routers

```
ping 10.1.1.1
ping 10.1.1.2
ping 10.1.2.1
ping 10.1.2.2
ping 10.1.3.1
ping 10.1.3.2
```

13) You should see the following type of response each time:

```
Spring#
Spring#ping 10.1.3.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.3.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/11/20 ms
Spring#
```

If you receive failure messages, you'll need to review your configuration and make necessary changes.