

Re: need help re. office network install

Source: <http://linux.derkeiler.com/Newsgroups/comp.os.linux.networking/2004-05/0754.html>

From: George Patton (george_m_patton_at_yahoo.com)

Date: 05/19/04

Date: Wed, 19 May 2004 00:09:29 -0500

Christopher Scott wrote:

- > *I'm looking for a little guidance/explanation on the ins and outs of*
- > *basic office networking. I know enough to understand what I'm talking*
- > *about but lack the experience to make a confident choice in this*
- > *situation...*
- >
- > *I'm currently working as the IT/developer for a small (20-person) firm*
- > *and their network is a mess, the result of years of neglect. They're*
- > *still using static IP, they have a gateway server w/ no special*
- > *firewall rules on it, they have a large DMZ that serves no purpose*
- > *(managed by the gateway) and are fronted by a Cisco router they can't*
- > *get access to (nobody has the password; I presume that this is*
- > *performing NAT).*
- >
- > *I've called in a few network install technicians to get some quotes and*
- > *they keep suggesting rather expensive (\$2000-3000) appliance devices -*
- > *clearly the ones they have the most personal experience with.*
- >
- > *What I'm starting to wonder is why can't I simply do this myself? Why*
- > *not just buy one or two switches and a Linksys router of some kind and*
- > *set it all up myself?*
- >
- > *As I mentioned, we have about 20-25 desktop machines that I want to*
- > *configure via DHCP services, 4-5 printers (which will require static IP*
- > *addresses), a file server (no outside access required) and a print*
- > *server (for the 4-5 printers). Web server and e-mail are both hosted by*
- > *an external service.*
- >
- > *I'm looking to implement a protective firewall, DHCP services and*
- > *possibly VPN access in the not-too-distant future. If I ever intend to*
- > *bring web and e-mail back in-house then I'll need port forwarding for*
- > *that, as well.*
- >
- > *Considering that we already own one 3Com Superstack 3 24-port switch*
- > *(and a slew of Baystack 255 hubs that I want to unload), I was*
- > *considering just getting one or two more Superstacks and a Linksys.*

- >
- > *Am I being foolish in thinking that something as small as a little blue*
- > *Linksys like what I have at home can be relied on in a small/medium*
- > *office environment? Is this too taxing an application for one of those?*
- > *And what about the difference between a BEFSR11 (the single-port blue*
- > *model) something like the RV016 or RV082? Considering that I've already*
- > *got switches then is there any advantage to having ports built into the*
- > *device?*
- >
- > *Any input, experience or suggestions would be greatly appreciated!*

Are you focusing on the internal traffic within the private network and simply need new switches to handle the traffic from one desktop to the next? Or, are you hoping to revamp the gateway and the firewall and the servers ASAP?

You haven't told us how the twenty people in your firm are using the internet, but if their needs are typical of most american companies I suppose I would do something like the following:

Identify the slowest machine in the company and make plans to re-task it as a linux firewall and NAT gateway. I would consider setting up one or two separate machines INSIDE the private network for use as a SAMBA file server, a DHCP server, and possibly as a print server running CUPS or LPRng. Finally it might be a good idea to plan on using one of the two samba machines for backup purposes. Depending on your needs, the backup device could be a relatively cheap DVD writer, or an older generation tape device (DDS3 or DDS4) or something much much larger.

As a general rule I would avoid using the gateway/firewall machine for any purpose other than a gateway/firewall. You could use it as for a backup DHCP server without too much risk but definitely don't use it for file serving, print serving, etc, etc. Otherwise you significantly increase your risk of compromise. Moreover I would urge you not to install anymore software on the firewall machine than is absolutely necessary. On most firewalls that I've installed there is very little stuff on the filesystem other than a custom kernel. Finally, it would be nice (down the road) to set up the firewall machine so that it boots from a read only cdrom drive and reads the firewall rule set from a write protected floppy. This effectively inhibits a would-be hacker from giving himself a permanent key to your establishment while still allowing you to easily modify the firewall rule set from day to day if necessary.

DAY 1 (ALL DAY)

Firewall/Gateway. You may be able to find a bootable cdrom image on the internet that already embodies a firewall application that will handle your immediate needs but if you take this course please make sure that you check out the credentials of the provider. And also run an md5checksum to make sure the image hasn't been tampered. Drawback: The

canned cdrom firewall images may not handle some custom requirements... such as masquerading VPN traffic and such, but they're definitely worth a look to get you up to speed fast. If you can't find a firewall cdrom image, you could install a minimal subset of the latest RH or Suse distro or whatever and use this to get started.

BTW, this technique of booting your firewall off a cdrom will require some planning on your part but a lot less time than learning how to modify the firewall rule set as time goes by, and it will significantly reduce headaches down the road. Apart from minimizing the damage if/when you're attacked, it will also make it easy to install a new firewall system if/when your existing machine dies from old age, lightning strikes, etc. [Insert cdrom in drive and hit <reset>.]

The firewall ruleset development task will also be significantly accelerated if you start with a decent script designed for building firewalls. You could probably have everything working pretty well initially within the first day. The big problem you'll encounter when you build a decent firewall is that you may encounter resistance from company employees who will resent the new ramifications of security. If you've never built a firewall before I would urge you to start with an extremely restrictive set of rules ---- perhaps allowing web browsing from within the company -- and then relax the rule set as needed if, and only if, your co-workers and boss convince you that they really need some new form of access. This way you won't have a panic-attack when you hear on the late night TV news that some internet chat program has a backdoor that russian gangsters have been using to examine commercial bank accounts of american businesses. Even still you should periodically "audit" your own system to make sure there aren't any open ports that you're unaware of. A friend of mine scoffed at the idea of a security audit on his system, but when he finally accepted my advice he found some glaring security flaws.

DAY 2 (AM)

The DHCP server will probably take you less than a couple of hours to set up on any of the latest stock distros. The stock documentation will include examples that you can simply re-write to suit your own needs. A configuration tool (webmin? perhaps) would make it even easier.

DAY 2 (PM)

A SAMBA file server is a bit more complicated to implement. Tools like SWAT and webmin will make configuration much easier at the get-go but down the road you should probably take a look at the actual configuration scripts so that you can fine tune them by hand if need be.

The biggest headache with SAMBA startups almost always involves authentication if/when you have an assortment of operating systems on your network. Even with the simplest form of security and authentication, Linux and Win98 and Win NT and Win 2000 all handle passwords a bit differently. My advice: Get things working in their

simplest form and add the complicated features after you get your feet wet. For starters, I would recommend that you go ahead and assign initial passwords for your users manually (coinciding with any windows passwords) so that their "home" shares on one or two linux SAMBA boxes will be accessible as soon as they log into their windows boxes. You can add the functionality for "syncing" passwords after you have the basics up and running.

DAY 3 (ALL DAY)

Printing. Print servers used to be easy to implement in the old AT&T lpadmin days until some newfangled concepts like IPP came along that were designed to make it easier to print from the internet. CUPS tackles some of the new protocols but also presents administrators with a whole slew of problems. A case in point would be a kernel bug in RH9 last year that would cause a machine to crash while printing to some HP network printers under certain circumstances. Hopefully your installation of a print server will proceed without any hitches but if you encounter problems there will probably be a relatively easy work around that won't cause you to tear your hair out. Stay flexible. Most people manage to configure the CUPS server for most office printers with relative ease in a relatively short amount of time... assuming that we're talking about a printer that sits on a desk and not a Xerox documate that occupies a medium size room.

DAY 4

You didn't mention backups, but if your network situation was truly neglected it's quite probable that the backups are non-existent or inferior and quite possibly useless altogether. I would strongly urge you to take a few hours — perhaps an entire day — and think through the procedures that have been established within your company for dealing with total disaster. IMHO, this is the most difficult part of your entire task. Commercial canned backup software may contain bugs that go unnoticed until disaster strikes, and this is only slightly less true in the case of open source tools like tar. Do your boss a favor and make sure that you have at least one decent backup device somewhere on your network and put it to good use on a regular basis. Your backup design chore isn't completed, BTW, until you try to do a "mock" restore on a brand new machine. You'd be surprised (horrified actually) if you knew how many times people try to restore their company archives and discover that some (possibly all) of the backups are useless.

General Advice

As with the firewall, I would urge you NOT to install anymore software on the samba and print servers than you absolutely need. It's easy to come back and add software later. It's significantly more difficult to delete software after it's been installed. If you want to play around with the latest and greatest gui desktops and games and such, you should install these on the machine in your office and NOT on your file server.

Apart from reducing security risks there are some other significant advantages to the minimalist approach. 1) Your servers will be faster and more productive if you don't have a lot of unnecessary crap running.

And your employer will appreciate the fact that your coworkers won't be complaining day and night about the "slowness" of the network. And you won't be hounding him/her to buy new motherboards with twin athlon 64 bit processors. 2) Your workload will be significantly easier. To illustrate this last point, a hobbyist won't have any qualms about spending the entire weekend installing the latest and greatest release of any given distro every six months on his relatively new Super Duper 10 megahertz MP gamer machine with three gigs of memory, but you can probably find better ways to spend your time when you're working with ordinary production hardware that stores and prints office documents all day long.

Let's say that you need SAMBA and DHCP and CUPS (for the print server).

If you install only these packages from a distro — ignoring the zillion other packages, your distro updates will be a piece of cake. Moreover, it will be a relatively simple job for you to install a compiler and libraries on one machine and then download the SAMBA and DHCP and CUPS sources from their respective repositories and compile and reinstall them yourself from time to time IF NECESSARY instead of waiting for the distro folks to release the latest kernel and the mega-set of binaries for the zillion programs that you don't really need. With the time and money you save by not having to install and update full sets of distros all the time, you can learn how to build custom kernels and speed up your lean and mean servers even more by stripping out deadwood in the typical kernel that you'll never (ever) need.