



Deploying Identity-Based Access Control

Peter J. Welcher

Introduction

Last month's article discussed technology, the 802.1x and EAP protocols, which can be used to authenticate users and grant or deny access to users of both wired and wireless networks. That article can be found at <http://www.netcraftsmen.net/welcher/papers/dot1x.html>.

This month, I'd like to take a look at what goes into actually implementing Identity-Based Access Controls. I'm currently involved in two college/university network design projects for which 802.1x is of interest. So I'm definitely practicing what I preach here!

Basic Orientation

This whole area is undergoing a lot of rapid change, so I'd like to start by covering the Big Picture. Cisco's top-level links tell the story. This whole area can be found under the title **Cisco Trust and Identity Management Solutions**, at the URL http://www.cisco.com/en/US/netsol/ns463/networking_solutions_package.html. That URL breaks this topic down into three areas, which I list below, along with the relevant links.

Identity (CiscoSecure ACS)	http://www.cisco.com/en/US/products/sw/secursw/ps5338/index.html
Identity-Based Network Services (IBNS)	http://www.cisco.com/en/US/netsol/ns340/ns394/ns171/ns75/networking_solutions_sub_solution_home.html
Network Access Control	http://www.cisco.com/en/US/netsol/ns466/networking_solutions_sub_solution_home.html

For Identity, Cisco's answer is the product, CiscoSecure ACS. This RADIUS and TACACS+ server software comes in Windows and UNIX variants, including a Service Provider product. It can use back-end user databases such as Novell, Windows NT, Active Directory, and LDAP.

For Identity-Based Network Services, IBNS, that's where a collection of products allow us to authenticate users via 802.1x and EAP, against CiscoSecure ACS and back-end user databases. RADIUS attributes allow us to then configure the network devices with per-user or per-user-group settings. The recent emphasis appears to be on dynamically-assigned VLAN's. For enterprises, think "employees", "on-campus-consultants#1", "guest", etc. For colleges, think "faculty and admin", "student", "server", "guest", etc. The idea is to issue VLAN's and corresponding addresses, perhaps from different address blocks. These may then provide the basis for all sorts of nifty policy actions, including access lists, routing, QoS, policy-based routing, etc.

I see some interesting design challenges lurking here, which we may get into in future articles. This definitely is venturing into new territory. What will scale and remain manageable is one of the big questions. The challenge is tying such user communities together across a Layer 3 core without getting seduced into large-scale Layer 2 VLAN's. MPLS VPN or VRF-Lite is one way to do that, a way that may be equally repulsive to some. L3 tunnels in general seem to what's currently under consideration to solve this.

Cisco lately has been talking a lot about Network Access Control, NAC. This definitely should be the subject of a future

article, perhaps my next one. NAC gives or will give the ability to deny access to the network not just based on user id but also based on whether the virus scanner or personal firewall software is running, how current the virus signatures are, etc. NAC is still evolving, with the first phases due to appear shortly. It's sounding to me like you may want to have not just a site license on the desktop virus software but also your vendor's central management software. NAC is founded on network equipment interacting with not only the client (as far as current state of a particular desktop) but also with security desktop management software, to verify that the client meets current security policy.

Identity-Based Access Control

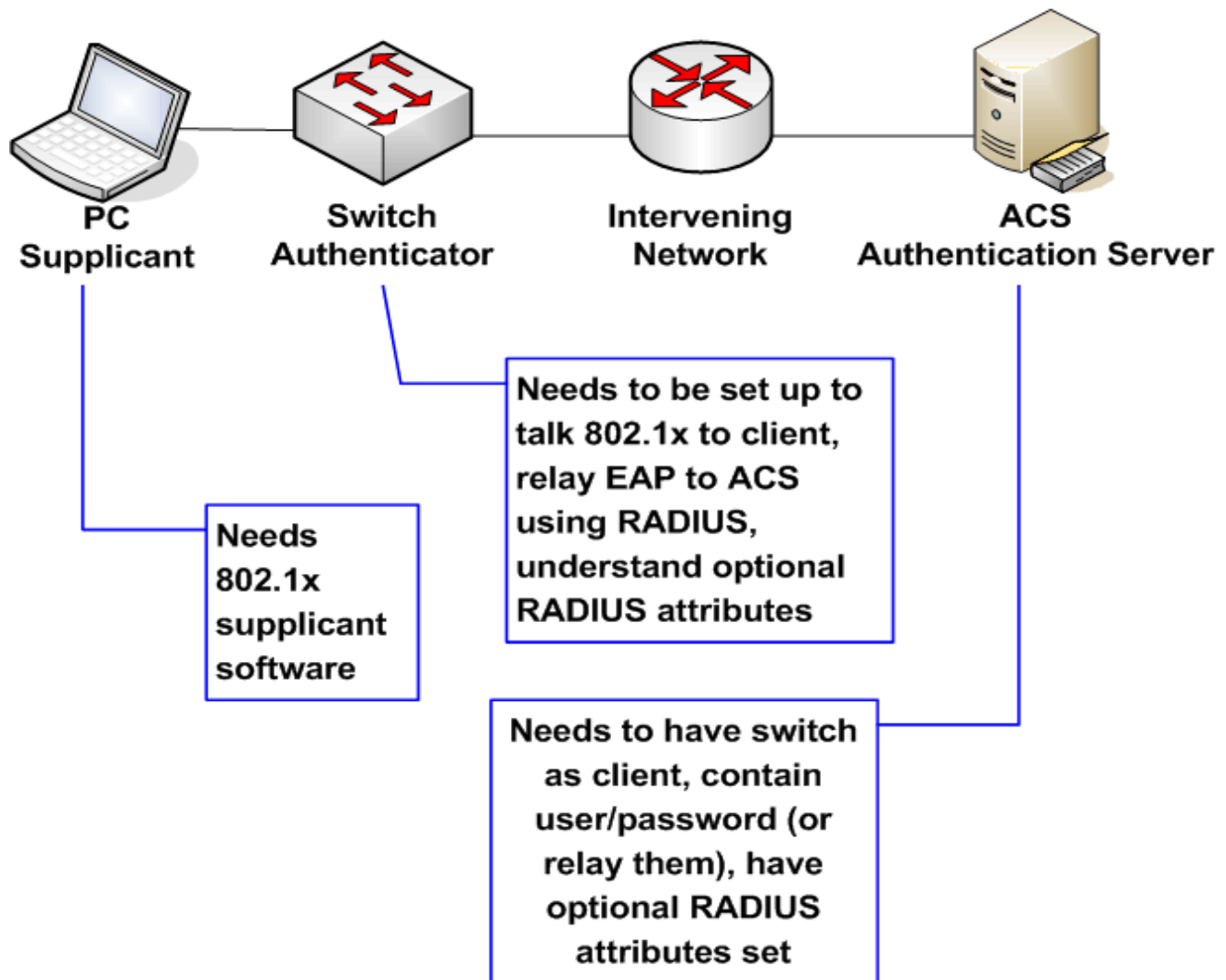
If you haven't guessed by now, this article fits firmly into the IBNS space. I want to give you an idea of what's entailed in getting IBNS working on a real network. Subsequent articles may drill down in more detail.

The ultimate good reference on all this right now appears to be titled **Network Infrastructure Identity-Based Network Access Control and Policy Enforcement Implementation Guide** (June, 2003). Yes, that's a mouthful! It can be found at http://www.cisco.com/application/pdf/en/us/guest/netsol/ns178/c649/ccmigration_09186a0080160229.pdf

I'm a big fan of doing lab work incrementally. Get something working, improve on it, gradually work your way towards what you want. That simplifies troubleshooting, since you're not trying to get 50 bazillion things working at the same time. For IBNS, my suggestion is to work with MD5 authentication first, even though that won't make use of a back-end database. After that, if you want to do PEAP, go for it. Tip: when installing MS certificate service, make sure you have MS IIS web server installed first. Otherwise you may have some head-scratching concerning the certificate services web interface and why it isn't working.

There are three areas requiring setup to get IBNS working. Plus arguably some follow-on activities.

The following diagram shows the main activities required for IBNS. You've got to install and configure client supplicant software on the desktop. The switches and WAP's have to be set up to require 802.1x authentication on appropriate ports. They also have to be set up to work with the ACS RADIUS server. The intervening network devices do have to provide connectivity between edge switches or WAP's and ACS. Then ACS needs to have the edge devices added as clients, to allow them to query it. ACS also needs to be set up with users and passwords, or set up to relay unknown user authentication actions to the appropriate back-end database. It is also helpful to set up ACS to handle groups of users, so you don't have to do a lot of repetitive work. If the back-end database uses user groups, they have to be mapped to ACS user groups.



There's also follow-up work. If you are doing dynamic VLAN's, they need to be created in the switches, and other switch configuration may need minor adjustment. They would also have to be setup in ACS via the appropriate optional RADIUS attributes. If these VLAN's are new subnets, you need to route them, you'll need DHCP scopes, you may need to adjust access lists, etc. We won't go into these latter details as (a) they'll vary from network to network, (b) this article would get too long, and (c) you'd probably find it boring anyway.

I hope I haven't scared anybody. This is mostly doing a little bit here, a little bit there, and getting it all right. ACS may take the most work, nothing that hard, but just a bunch of getting it set up to work with your devices if you haven't already done that.

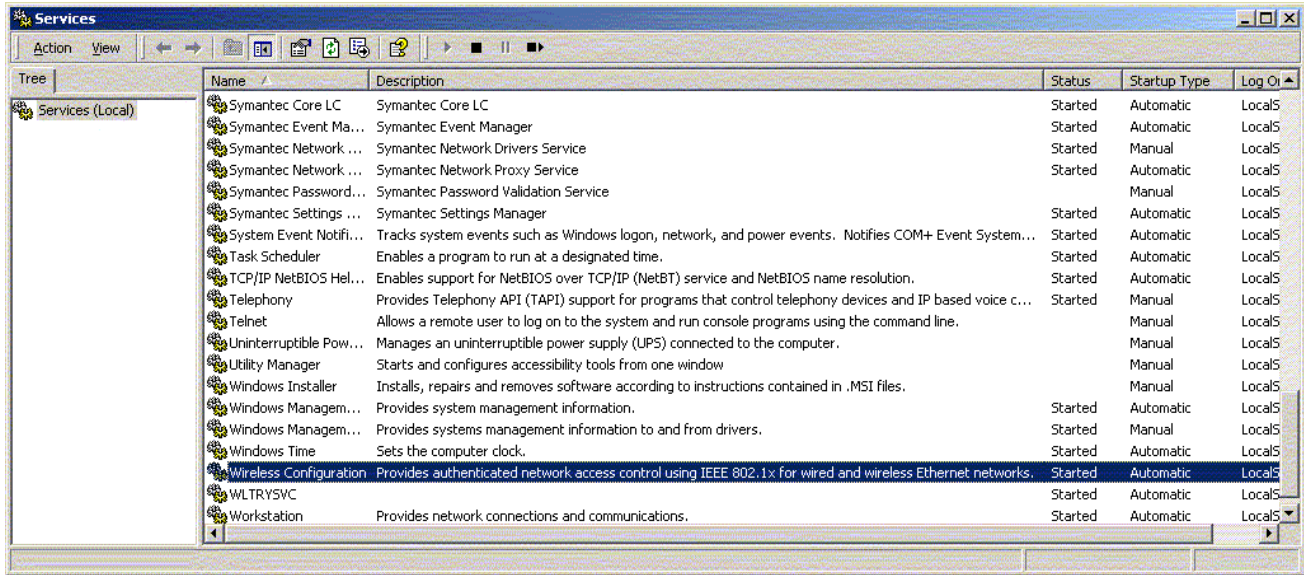
Let's now take a brief look at what's involved for each of the above 3 areas: supplicant, client edge device, and ACS.

Supplicant

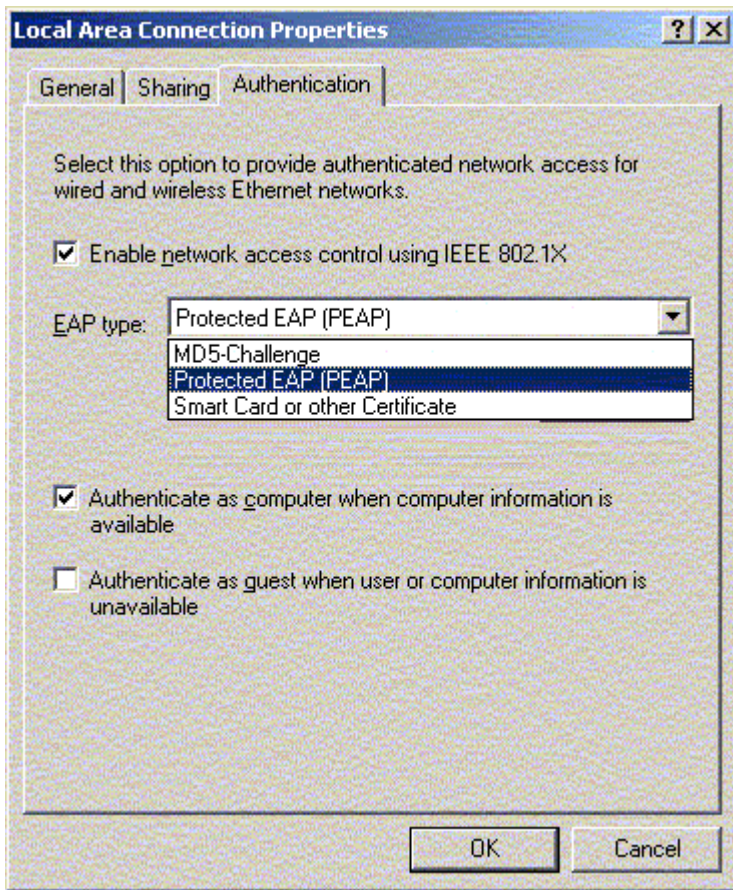
The following screen shots are from my PC. I'm running Windows 2000 Pro, SP4. That includes the SP3 support for 802.1x. The little gotcha is that you do have to enable the 802.1x support, which is as a service. My understanding is that the service listens for an 802.1x request and then prompts the user for username/password/domain. Another gotcha is that it doesn't seem to automatically start for my wired port unless Ethernet link state is detected while booting (probably at service startup time).

The Microsoft Knowledge database did help me solve the first of the above gotchas. Some head-scratching did ensue, since the article helpfully never mentioned the name of the service to start. It's not named anything obvious, like dot1x or 802.1x Service. In fact, see the below figure, it's named "Wireless Configuration". Since I already had 802.1x working on

my wireless side without enabling this (via the Dell driver), I didn't catch on until I expanded the description column as shown and saw "wired". Ok, maybe that was obvious to you. It wasn't for me!



You then set up the appropriate interface for authentication as shown in the next figure. The new authentication tab appears after you've started the above service. I mention this because it strikes me as rather counter-intuitive.



Pick your authentication technique and you're good to go. MD5 is simplest, but requires setting up a user/password in CiscoSecure ACS (Microsoft back-end databases don't expose MD5 adequately for ACS to verify login).

Aside from the technical details, the main issues with 802.1x supplicant seem to be (a) obtaining one, and (b) user policy. Neither is necessarily particularly hard.

If one has a wireless card, up-to-date drivers probably take care of authentication support. Windows XP has 802.1x support. Windows 2000 SP3 or later includes 802.1x support, as noted. I understand free client support is available for Linux and MacOS. That leaves out legacy devices (printers, other UNIX variants, etc.). It also leaves out older versions of Windows. But Funk and Meetinghouse both have driver support available for older Windows PC's. One can download it from their websites for around \$40-50 per PC.

The policy issue arises with colleges because of hesitancy, either on cost or support. Both can be complex issues.

Re cost, it costs money to upgrade Windows, or to buy the Funk or Meetinghouse driver. I've got two kids in college, I understand how sensitive parents are about cost. Colleges are very hesitant to dictate more cost for their students. Part of that goes with being an open environment. Part of it may also relate to staying competitive. Operating systems upgrades may not be feasible, since they may require the cost of a new PC. That's a show-stopper for many families. Concerning purchasing drivers for older Windows variants, one could argue that \$40 per student is cheap, compared to the support costs of worm containment. That's where policy becomes a problem. Some colleges feel they cannot afford to support the wide variety of desktops or the sheer number of PC's their students have. And installing drivers or requiring installation of drivers means you own any support problems the student subsequently encounters. So do you provide "legacy support" (see below)? Is a component of it lesser connectivity as an incentive for students to opt for the buy-and-install-drivers solution?

One counter-argument is that students who get virii do impose very real costs on the college. If they get a worm that creates traffic, it may adversely affect other users or even knock out the network.

Colleges often try to provide anti-viral software to their students. Who then ignore it, don't install it, turn it off, etc. The necessary teeth might then be access control (NAC? web-based?), or it might be a hefty fine if your PC gets infected and starts trashing the network. I personally think stringent standards up front ("you must have one of the following anti-viral products installed") with fee for non-compliance may be where we all end up. But even there, how does one induce the unwilling to periodically update their virus signatures and refrain from turning the protection off? That's where NAC will be of intense interest to colleges.

This is also why 802.1x arises in college environments right now. Student PC's are a very scary unknown right now, not under any anti-viral control. So it is highly desirable to identify student machines and isolate them (quarantine them?) in student VLAN's. Then run traffic from those VLAN's through firewalls or IDS's. That at least addresses damage containment now.

Having a focal IDS allows identification and treatment of infected PC's. Down the road, NAC may provide more enforcement without major labor burden. At that point, maybe student PC's become trusted again, and the policy split becomes "NAC-approved" versus "guest and non-NAC-approved PC's".

To sum that up, what's messy with clients right now is figuring out what is technically feasible, and what the policy ought to be. The interaction with culture and expectations of the surrounding environment make this particularly sensitive for colleges.

In a few years, most people will be running OS variants that support 802.1x. No doubt we'll all have some other hot issue then. In the meantime, we do have the challenge of transition and "legacy support". Either users have to have OS upgraded, have to have supplicants installed, or there has to be some way to support legacy non-802.1x devices. The latter is another potential article. I've got two techniques that you may find useful, ones that work right now. Which to use depends on your policy and needs.

Client Edge Device

Here's most of the captured configuration from the Cisco 3750 I was using to test all this with. It's pretty basic. One very useful option is to set up a guest VLAN, and use that as the default for those who don't successfully authenticate via 802.1x. That seems like a useful topic, but perhaps one for the "legacy support" article.

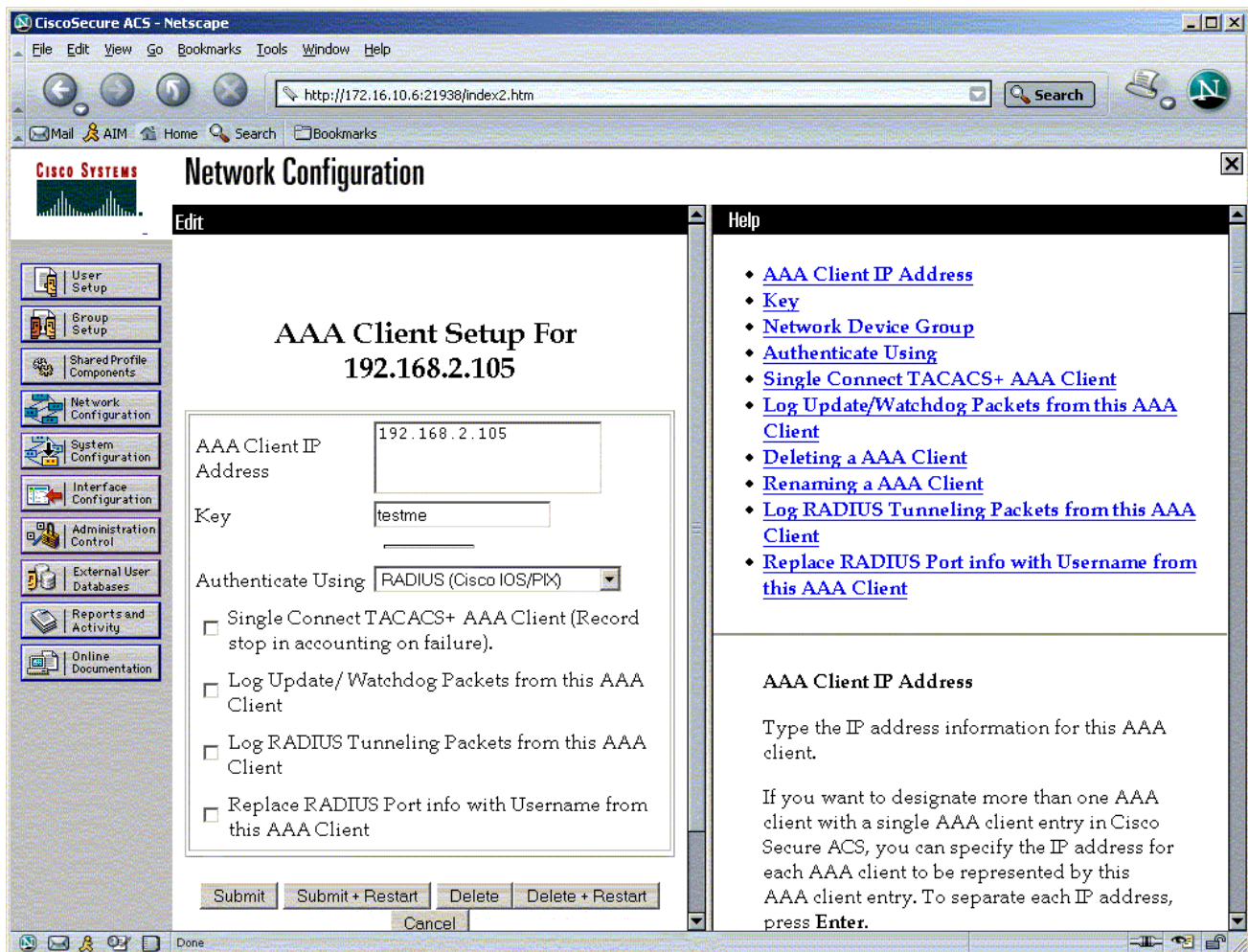
```
hostname c3750
!
aaa new-model
aaa group server radius radserv
    server 172.16.10.6 auth-port 1645 acct-port 1646
!
```

```
aaa authentication login default line enable
aaa authentication dot1x default group radserv none
aaa authorization config-commands
enable password cisco
!
ip subnet-zero
ip routing
!
interface FastEthernet1/0/3
 switchport mode access
 no ip address
 dot1x port-control auto
 spanning-tree portfast
!
interface Vlan1
 ip address 192.168.2.103 255.255.255.0
!
router eigrp 1
 network 192.168.2.0
 auto-summary
!
ip classless
!
radius-server host 172.16.10.6 auth-port 1645 acct-port 1646 key trustme
radius-server key trustme
```

I left the addresses in to show that this really was working across an intervening router. The ACS server was at 172.16.10.6.

CiscoSecure ACS

The first key thing here is to make sure ACS knows about the client edge device and will respond to it. See the following screen capture. A client is added by clicking on Network Configuration and then the Add Client button.



There are a good many more options to configuring ACS, but it feels now that they properly belong in an article about just ACS. So I'll save those for another time.

Troubleshooting 802.1x

"Debug dot all" and "debug radius" are two very useful switch commands. I have captures I'd like to include, but no room. The idea is, see if the switch and PC talk, see if you get the login dialog box on the PC, see if that data gets to the switch, and then see if the RADIUS exchange takes place with correct results. If your RADIUS configuration in the switch is wrong, you're either not going to send RADIUS requests, or not going to receive replies.

I also found the ACS reports on passed and failed authentications very useful. The failures often told what the problem was and how to fix it!

Summary

For details, especially if you get stuck, do see the URL mentioned above, http://www.cisco.com/application/pdf/en/us/guest/netso/ns178/c649/ccmigration_09186a0080160229.pdf.

There was also a good Networkers 2003 presentation on this topic, complete with many ACS screen captures. See <http://www.cisco.com/networkers/nw03/presos/docs/SEC-2005.pdf>.

I've decided to go ahead and post the ACS screen captures I have, pending writing an article on ACS. They are intended more as a visual tour of ACS than as showing the specifics of how to set up for 802.1x, PEAP, dynamic VLAN's, etc. See <http://www.netcraftsmen.net/welcher/papers/acs-3.2-cap.pdf>.

Possible upcoming topics: Troubleshooting 802.1x. Dynamic VLAN's via 802.1x (also known as "How I Really Did It"?). CiscoSecure ACS. NAC. PKI: MS Active Directory and router IOS-based PKI, what networking folks need to know. Alternatives when 802.1x isn't the answer ("legacy support"). WAP authentication configuration. Cisco Wireless Domain Services (WDS), which help scale and cache wireless authentication information and RF reporting.

Let me know what you think of those topics!

Dr. Peter J. Welcher (CCIE #1773, CCSI #94014, CCIP) is a Senior Consultant with Chesapeake NetCraftsmen. NetCraftsmen is a high-end consulting firm and Cisco Premier Partner dedicated to quality consulting and knowledge transfer. NetCraftsmen has eight CCIE's, with expertise including large network high-availability routing/switching and design, VoIP, QoS, MPLS, IPSec VPN, wireless LAN and bridging, network management, security, IP multicast, and other areas. See <http://www.netcraftsmen.net> for more information about NetCraftsmen. Pete's links start at <http://www.netcraftsmen.net/welcher>. New articles will be posted under the Articles link. Questions, suggestions for articles, etc. can be sent to **[pjw <at> netcraftsmen <dot> net](mailto:pjw@netcraftsmen.net)**.

5/2/2004

Copyright (C) 2004 Peter J. Welcher