

INTERESTED IN CONTROLLING A COMPUTER DOWN THE STREET OR HALFWAY AROUND THE WORLD? HOW ABOUT SWITCHING AN APPLIANCE'S POWER, DIMMING THE LIGHTS, OR ADJUSTING AMBIENT TEMPERATURE, ALONG WITH ATTAINING SOME AUDIO-VISUAL FEEDBACK ON A PROPERTY'S STATUS? IT'S ALL POSSIBLE, ALBEIT NOT WITHOUT GLITCHES.



Homeland. security:

MONITORING AND MANIPULATING REMOTE RESIDENCES

Figure 1 This hands-on project's test bed has some particularly challenging attributes: a remote location and inconsistent power.

BY BRIAN DIPERT • SENIOR TECHNICAL EDITOR



Ubiquitous network protocols, such as IP (Internet Protocol), TCP (Transmission Control Protocol), and UDP (User Datagram Protocol), in combination with the increased availability and decreased cost of robust broadband-Internet access, have cultivated an upsurge in remote-access and -management capabilities. Numerous technologies exist, for example, to control computers from outside a LAN (local-area network)—from the modest but omnipresent RealVNC (virtual-network-connection) to VPN (virtual-private-network) applications, such as LogMeIn's freeware-plus Hamachi.

Operating-system-specific remote-computer-control variants include Microsoft's Remote Desktop Connection and Apple's Remote Desktop. And, with products such as Sling Media's Slingbox and Microsoft's WebGuide for Windows XP Media Center Edition and Windows Vista, you can peruse live and archived video material from anywhere in the world (see "Homeland security: remotely tune into TV content" at www.edn.com/briansbrain).

This EDN hands-on project testdrives the reality behind these concepts' theo-

ries. Additionally, it tackles video surveillance, along with a hands-on evaluation of the home-automation technologies that EDN previously covered (**Reference 1**). However, whereas the earlier article's focus was on within-the-home control, I've built on its foundation by additionally attempting to control—and monitor—a home from the outside. The study's test bed is a diminutive dwelling built in the mid-1980s, located atop a 7000-foot ridge in the Sierra Nevada mountains and prone to fairly frequent rain-, snow-, and wind-induced power loss of random duration (**Figure 1**).

Although this article's analysis focuses on a secondary residence, if you brainstorm for a few minutes you'll likely come up with a lengthy list of additional uses for this project's results. Consider, for example, a remote office suite whose equipment, illumination, and temperature you might want to be able to remotely monitor and adjust. Assuming that the equipment, along with data stored on hard drives, on optical discs, and in file cabinets, is of substantial value, you might also be motivated to inspect the premises from afar and respond to sensor alerts of a fire, a break-in, or another catastrophe. And don't underestimate this article's applicability to home-based-health-care trends (**Reference 2**). If, for example, a friend or family member stricken with Alzheimer's disease were to wander out the door, wouldn't an e-mail- or pager-based alert be useful?

The journal of my experiences in the following paragraphs will, I hope, smooth the path for those of you inter-

AT A GLANCE

▣ Determining your dynamic-IP (Internet Protocol)-based Internet service's address is more complicated than it might appear at first glance.

▣ Webcams are prone to unreliability due to temperature and wireless-connectivity variations. Watch out, too, for integrated Web servers that unnecessarily restrict you to one browser or operating system.

▣ Power-line-based control schemes have tremendous potential but, my testing suggests, aren't yet robust enough for widespread adoption. Monitor the *Brian's Brain* blog at www.edn.com/briansbrain for ongoing updates as I continue my Insteon debugging and begin exploring the Z-Wave wireless alternative.

ested in following my footsteps. Equally important, I hope that those of you creating technologies and designing products based on them for germane applications will optimize them using this write-up's observations and conclusions as a guide. Remember: If, as an engineer, I struggle with a given technology, the average consumer has even less chance of figuring it out.

DNS ANGST

In researching before the eventual purchase of my mountain getaway, a two-hour drive away from my primary residence at the time, I was pleasantly surprised to find that this second home could access both cable and DSL (digital-

subscriber-line) broadband service. An abundance of competing Internet-access options isn't always available, but satellite-Internet service—albeit with the requisite long latency, weather-dependent uptime, performance irregularity, and relatively high cost—should at minimum be an available broadband candidate, as long as you have an unobstructed view of the sky above your dwelling (**Reference 3**). Most broadband providers offer “base” services that employ dynamic IP-address allocation. After a provider-dependent WAN (wide-area-network)-inactivity time period has elapsed, your LAN's allocated IP address releases and returns to an available “pool.” The next time any portion of your network connects to the Internet, it's statistically likely that your service provider will assign your LAN a different IP address.

Dynamic allocation precludes the possibility of reliable IP-address-based access to your LAN. Some broadband providers offer their customers optional static-IP addresses but usually at a significant price premium reflecting their “business-class” status. Instead of paying AT&T extra cash for a static-IP address, I evaluated several of the DDNS (dynamic-domain-name-system) services that attempt to address the issue. They work with equipment, such as a router or a computer, within your LAN. When the equipment senses a change in your service-provider-allocated IP address, it automatically connects to the DDNS provider's server and updates your account information. By using the DDNS account-allocated URL (uniform-resource locator) and as-



Figure 2 The DDNS client in D-Link's DI-524 appears to be nonfunctional (a), and the one in Linksys' WRT54GC is intolerant of lengthy network-initialization sequences (b). DynDNS' client software running on a Fujitsu Lifebook P-2110, however, operates as intended (c).

suming that the Internet-service provider you're currently connected to has an up-to-date DNS server, you'll always find your remote LAN regardless of what its IP address is.

The first router I tried, D-Link's DI-524, claims to support DDNS, but DNS-service provider No-IP refused to acknowledge address updates that the DI-524 sent (**Figure 2**). After some research, I discerned that No-IP was likely blocking the update attempts because D-Link's DDNS client historically acts aggressively (**Reference 4**). I couldn't find any reference to the update-server addresses for DynDNS and TZO (Tzolkin), so I couldn't test the DI-524 with either of these two services. Linksys' WRT54GC router also integrates a DDNS client, in this case with built-in support for DynDNS and TZO. However, although the WRT54GC DDNS client generally worked better than the one in the DI-524, its shortcomings still rendered it unusable for my setup.

When my Siemens SpeedStream 4100 B DSL modem initially connects to AT&T's network, it can take a minute or more for the modem—and, therefore, the router—to receive a dynamic-IP-address assignment. My Linksys OGV200 QOS (quality-of-service) network optimizer's autocalibration cycle further extends the delay until initialization is complete. Unfortunately, the WRT54GC's DDNS client seemingly is too unintelligent to handle this deferral. If it is initially unsuccessful at logging into DynDNS' and TZO's update servers, it reports an "error in username or password," "unable to establish HTTP connection," or similar message indicating lack of connection success and doesn't reattempt a later login.

Because I'm powering the DSL modem, QOS processor, and router from a battery-backed UPS (uninterruptible-power supply), you might think this glitch would be easily solvable with a one-time manual login from the router's browser-based GUI (graphical user interface). After all, once the WRT54GC successfully logs into a DDNS server, it adequately handles dynamic-IP-address updates. On at least one occasion, however, the premises' power loss was sufficiently long to completely drain the UPS battery, thereby shutting off all of the network gear. When the premises'



(a)



(b)



(c)



(d)

Figure 3 D-Link's DCS-1000W, being Java-based, is browser- and operating-system-agnostic but can't handle sun exposure (a). The 802.11b-based DCS-5300W (b) and 802.11g-supportive DCS-5300G (c) were unreliable over Wi-Fi, but Actiontec HomePlug AV adapters practically solved the connectivity problems (d).

power came back on, the router once again gave up after unsuccessfully attempting to log into my DynDNS and TZO accounts; therefore, I completely lost access to the LAN until I returned to the residence several weeks later.

Ironically, another piece of active equipment on my LAN, the VOIP (voice-over-Internet Protocol) adapter, *also* automatically—and, at least so far, always successfully—logs into a WAN-based server. The IP address associated with my VOIP account is therefore identical to the AT&T-allocated dynamic-IP address for my LAN at any point in time. Unfortunately, neither BroadVoice nor Vonage allows customers to access this information. I plan to eventually test Linksys' WRT54GL router, both with factory-supplied firmware and with DDNS client-inclusive open-source code, such as DD-WRT and Tomato. For now, though, I've resorted to installing DynDNS' client software on a power-efficient Fujitsu Lifebook P-2110 laptop computer. The PC is old and slow, based on an 867-MHz Transmeta Crusoe CPU, but my application isn't performance-critical. To maximize the laptop's probability of surviving a lengthy power loss, I've connected it to both an extended-capacity main battery and an optical-drive-bay-based supplemental battery, with further help from an APC (American Power Conversion) external universal-notebook battery. The environmentalist in me isn't thrilled with the idea of an always-on computer, but, as you'll soon see, I've also found another use for the system.

CONVOLUTED VOYEURISM

Once I figured out how to reliably contact my router from the WAN, the next step was to open up firewall holes so that I could access the LAN gear behind it. I wanted to set up two Webcams, one pointed out the front door—to monitor, among other things, wintertime snow conditions—and the other perusing the home's interior. The first camera I tried, D-Link's now-obsolete, Java-based, 802.11b- and Category 5-cable-supportive DCS-1000W, was browser- and operating-system-agnostic—giving it advantages over ActiveX-based alternatives that you'll soon read about (Figure 3). Unlike its successors, it offered no movable-lens capability, and



Figure 4 Smarthome's 2412S PowerLinc Modem is outlet-sensitive, at least in my setup, and it must be fully powered before you power up Universal Devices' ISY-26 to reliably manage it (a). The Smarthome 2443 access points aren't seemingly acting as the wireless phase couplers they're intended to be, although the reason why is unclear (b). Once I get my Insteon network to a more robust state, I'll advance it beyond its current humble implementation, switching two incandescent lamps via Smarthome 2856S3B on/off adapters (c).

it couldn't take the heat in the aptly named sunroom where I installed it. As spring turned into summer, the DCS-1000W began—after a random period of stable operation—ignoring network-access requests until I cycled its power, a difficult task when I was off-site.

My setup now consists of two D-Link DCS-5300 Webcams, both supporting 10/100-Mbps Category 5-cable connections. The DCS-5300W variant is 802.11b-cognizant, whereas its G twin handles higher bandwidth 802.11g. Achieving success was a diagnostic struggle; by the end, the wireless-protocol differences between the two cameras were irrelevant. At first, the no-wires appeal of 802.11 encouraged me to go in that direction. However, despite having no discernible contending 2.4-GHz interference within the premises and using a broadcast channel that didn't overlap any of the nearby neighbors' faint Wi-Fi beacons, any wireless connection I established between the router and either Webcam survived for no more than a few days.

At first, I thought that DHCP (dynamic-host-configuration-protocol) renewals were failing, so I configured both Webcams with static-IP addresses. Although this tack is always a good idea because it provides a stable forwarding destination for firewall holes, it didn't provide any discernible reliability relief in my case. I never figured out the root cause of the wireless-connectivity problem. Were the D-Link Webcams or the D-Link and Linksys routers to blame, was it some nuance of the interaction between them, or could it be as-yet-undetected environmental interference? Not relishing the idea of crawling under the house to run cable through floors and walls, I instead used Actiontec's HomePlug AV adapters, which so far have been generally reliable in operation—albeit with a few hiccups—and deliver discernible audio through the Webcams' built-in microphones and smooth video over UDP (**Reference 5**).

The DCS-5300W and DCS-5300G employ ActiveX-based video add-ins, meaning that—unlike with their Java-cognizant DCS-1000W predecessor—if you attempt to use a browser interface to view the vistas they capture, you can do so only from a Windows-based computer and only from Internet Explorer.

Even with those restrictions in mind, I found that two of the four Windows XP-based systems I regularly access cannot view Webcam content from Internet Explorer; I get “HTTP-400-bad-request” errors whenever I make an attempt. I'm more successful in achieving this access using Firefox's IE Tab add-on, which runs the Internet Explorer rendering engine. In this case, I can log onto the Webcams, and, if I repeatedly refresh each frame within a given DCS-5300 Web page, its content will eventually appear.

Neither other folks' PCs nor my other two Windows systems experience the same problem. I suspect that some other installed Internet Explorer add-on is causing a conflict, although I've tried disabling all of the obvious candidates with no effect, or perhaps an obscure Windows-security setting is to blame. Fortunately, D-Link's D-ViewCam application works on every Windows system I've tried it on, thereby providing an alternative access path. Other DCS-5300 grumbles include its somewhat-noisy operation and lack of optical-zoom capability. I like the Webcams' pan-and-tilt feature, however, and I should also note that I haven't yet employed some of the application's advanced features, such as the ability to detect and react to motion, to interface with external sensors, and to periodically—and in response to a trigger event—e-mail images and copy them to an FTP (file-transfer-protocol) server.

CONTROL ISSUES

Except for infrared units, Webcams work only when adequate ambient light allows the devices to capture a meaningful image. For example, if a burglar were prowling around in my home after dark, the camera for viewing the interior would be useless unless the lights were on. Given my “green” leanings, keeping the lights on for hours or days at a time with nobody home is an unappealing solution. This quandary explains one of my motivations for adding WAN-accessible home control to the technology mix: It also would be nice to keep the thermostat low when I'm away and ramp it up from afar a few hours before I return home.

Although HomeSeer Technologies and its partner, Cooper Wiring Devices,

have supplied me with some Z-Wave-based wireless equipment, I've focused my near-term attention on power-line-control technologies. Part of the reason for this power-line prioritization is my earlier-described frustrations with Wi-Fi. Initially, I planned to tackle X10, spurred on by an excellent reference manual (**Reference 6**). This path was appealing given the wealth of new, often discontinued, and barely used X10 equipment that eBay and other sites offer. In retrospect, this abundance portends potential problems as much as cost-effectiveness. You have to wonder why so much barely used gear is available for purchase.

After a consultation with Smarthome, I became aware of X10's substantial shortcomings, particularly the lack of guaranteed feedback to a control-transition request. Smarthome's literature refers to it as "unacknowledged, 'press-and-pray' signaling" (**Reference 7**). This lack of feedback would be especially problematic if you were attempting to manipulate a remote setup. Instead, I've been experimenting the past few weeks with Insteon technology, which builds on an X10 foundation, from Smarthome's parent company, SmartLabs. For example, the company claims X10 compatibility with Smarthome's model 2412S PowerLinc Insteon modem, which I currently use (**Figure 4**). Before proceeding down the Insteon path, I obtained assurances from Smarthome and HomePlug technology developer Intellon that an Insteon control network would cohabit—although not communicate—with my HomePlug AV setup.


Speaking of HomePlug, I frankly feel like I'm back in the painful days of HomePlug 1.0 as I strive to get the Insteon setup working stably (**Reference 8**). To date, in the spirit of crawling before walking before running, I've attempted to control only two 2856S3B on/off adapters connected to nondimming incandescent lamps. The 2412S can "see" neither of the adapters from two of the three power outlets I've tried

connecting it to, even though the adapters are in the same room as the modem and less than 10 feet away from each other. Proximity is fairly meaningless when it comes to power grids; nearby 110V outlets may come from different circuit breakers or, even worse, be on opposite phases of the 220V source feed. However, the model 2443 access points I also have installed are supposed to, by RF-linking to each other, bridge Insteon control signals across the two 110V-ac phases. This bridging doesn't seem to be happening, and these setbacks are bothering me because this small, modern home shouldn't present much of a problem to a robust power-line-control technology.

Much of today's home-control equipment, such as the 2412S and its sibling 2414S PowerLinc Controller, still relies on the archaic RS-232 interface. For stand-alone—that is, not PC-based—operation, I interfaced the 2412S to Universal Devices' Category 5- and RS-232-inclusive ISY-26 home-automation controller, which embeds a Java-based, albeit nonintuitive, Web-server user interface. The combination worked fairly well as long as I plugged the 2412S into the correct ac outlet and ensured that the 2412S was fully operational before powering up the ISY-26. In an unstaffed remote environment prone to frequent power loss, this precise power-sequence requirement is untenable. Although the ISY-26 literature clearly documents the sequencing constraint, my Universal Devices contact assures me that in real-life practice it's not necessary. He suspects that I have either a faulty 2412S or an ISY-26 with out-of-date, bug-prone firmware. I'll continue experimenting until press time; monitor my blog for further progress reports.

After encountering the ISY-26's limitations, I also attempted to manage the 2412S from the Fujitsu Lifebook P-2110 laptop with HomeSeer's HS2 home-control software and an SIIG model JU-HS2012-S2 USB-to-dual-RS-232 adapter. Again, as long as I used the correct ac outlet, this combination worked fairly well after I overrode the default, nonfunctional drivers that Windows XP auto-installed when I first plugged the JU-HS2012-S2 into a laptop USB port. I wouldn't recommend exposing the HomeSeer software's Web-server

MORE AT EDN.COM 

 Go to www.edn.com/071122df for this article's associated vendor box or to post a comment to our Feedback Loop.

interface to the WAN through a firewall hole, however, especially over the default HTTP port 80. Two days after I took these very steps, the laptop stopped responding to WAN-access attempts. When I returned on-site nearly two weeks later, I found the system locked up with a blank screen. Power cycling the PC brought it back to life with no apparent ill effects, so I suspect that someone unsuccessfully attempted to hack it. Nevertheless, use a nonstandard TCP port or, better yet, dispense with the direct Web-server interface and instead access the HomeSeer-equipped computer over an encrypted and password-protected VNC or VPN connection, as I'm now doing. **EDN**

REFERENCES

- 1 Quinell, Richard A, "Networking moves to home automation," *EDN*, July 5, 2007, pg 40, www.edn.com/article/CA6455597.
- 2 Dipert, Brian, "The human touch keeps the elderly and the disabled technology-connected," *EDN*, Dec 17, 2004, pg 47, www.edn.com/article/CA486571.
- 3 Dipert, Brian, "Satellite-served Internet: so slow," April 30, 2006, www.edn.com/blog/400000040/post/1970003197.html.
- 4 No-IP, <http://blog.no-ip.com>.
- 5 Dipert, Brian, "Home transportation: benchmarking power line, 802.11, and Ethernet," Aug 2, 2007, pg 40, www.edn.com/article/CA6462560.
- 6 Meyer, Gordon, *Smart Home Hacks: Tips and Tools for Automating Your House*, ISBN 10: 0-596-00722-1 ISBN 13: 9780596007225, O'Reilly Media, October 2004, www.oreilly.com/catalog/smarthomehks.
- 7 "Insteon Compared," Smart Labs Technology, Jan 2, 2006, [www.smartlabsinc.com/files/INSTEON Compared20060102a.pdf](http://www.smartlabsinc.com/files/INSTEON%20Compared20060102a.pdf).
- 8 Dipert, Brian, "A man, a LAN, a plan," *EDN*, Aug 19, 2004, pg 22, www.edn.com/article/CA443377.

You can reach
Senior Technical Editor
Brian Dipert
at 1-916-760-0159,
bdipert@edn.com,
and www.bdipert.com.

