The Internet Protocol Journal

March 2004 Volume 7, Number 1

A Quarterly Technical Publication for Internet and Intranet Professionals

In This Issue

You can download IPJ back issues and find subscription information at: **www.cisco.com/ipj**

From The Editor

The operational stability of the global Internet (or any network based on TCP/IP technology) is in large part the result of a carefully configured routing system. Routing continues to be one of the most complex topics in Internet engineering. In our first article, Russ White describes some mechanisms for the design of large-scale, stable routing systems. The article is entitled "High Availability in Routing."

Security continues to be a high-priority item in computer networks and in society in general. One aspect of security is the identification system by which an individual is given authorized access to a particular facility, be it physical or virtual. Edgar Danielyan gives us an overview of one key element of identification, namely *biometrics*.

The Internet is "going where no network has gone before." The *National Aeronautics and Space Administration* (NASA) has been working on the *Interplanetary Internet Project* (**http://www.ipnsig.org/**). We hope to bring you an in-depth article about this project in a future issue. An important demonstration of this system took place recently. To quote from the press release:

> "A pioneering demonstration of communications between NASA's Mars Exploration Rover *Spirit* and the *European Space Agency* (ESA) *Mars Express* orbiter has succeeded. On February 6, 2004, while Mars Express was flying over the area Spirit was examining, the orbiter transferred commands from Earth to the rover and relayed data from the robotic explorer back to Earth. The commands for the rover were transferred from Spirit's operations team at NASA's *Jet Propulsion Laboratory* (JPL), in Pasadena, California, to ESA's European Space Operations Centre in Darmstadt, Germany, where they were translated into commands for Mars Express. The translated commands were transmitted to Mars Express, which used them to successfully command Spirit. Spirit used its ultra-high frequency antenna to transit telemetry information to Mars Express. The orbiter relayed the data back to JPL, via the European Space Operations Centre."

We often receive requests for back issues of IPJ. Although we cannot provide paper copies, all of our previously published editions are available in both PDF and HTML format from the IPJ Website: **www.cisco.com/ipj**.

—Ole J. Jacobsen, Editor and Publisher
ole@cisco.com

High Availability in Routing

by Russ White, Cisco Systems

A network is a complex system of interacting pieces, as anyone who has ever worked with a large-scale network "in the wild" can tell you. So, when businesses begin asking for a network that can converge in something under 1 second, especially in a large network, network engineers begin to scratch their heads, and wonder what their counterparts in the business world are thinking about. Just about everyone in the network engineering business knows scale and speed are, generally speaking, contradictory goals. The faster a network converges, the less stable it is likely to be; fast reactions to changes in the network topology tend to create positive feedback loops that result in a network that simply will not converge.

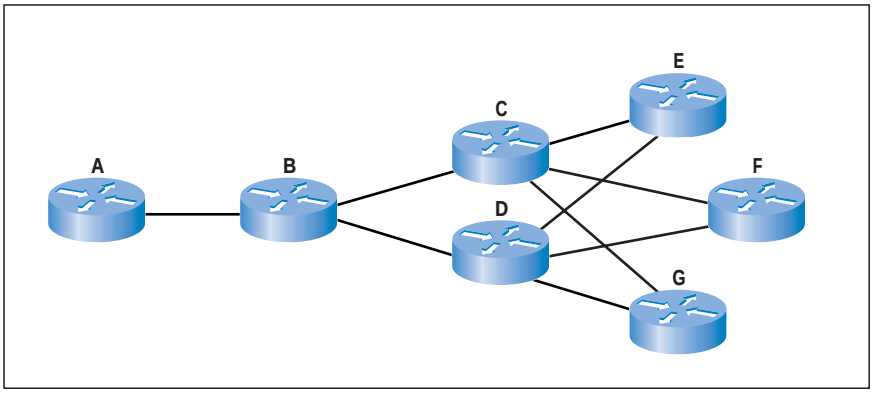
But recent experience has shown that subsecond convergence in a network—even a large network in the wild—is definitely possible. How do we go about building a large-scale network that can converge in times that were, before recently, considered impossible, or improbable, at best? We approach the problem the same way network systems, themselves, are approached. We break the problem down into smaller pieces, and try to solve each piece individually. When we have solved each of the smaller pieces, we recombine them, and see what needs to be adjusted to make it all work together properly.

What pieces of a network do we need to be concerned about when considering subsecond (fast) convergence? Generally, we are concerned with the physical layer (how fast can a down link be detected?), routing protocol convergence (how fast can a routing protocol react to the topology change?), and finally, forwarding (how fast can the forwarding engine on each router in the network adjust to the new paths calculated by the routing protocol?). This article focuses on routing protocols convergence, with some discussion of fast down detection as well, specifically the interior gateway protocols, *Enhanced Interior Gateway Routing Protocol* (EIGRP), *Intermediate System-to-Intermediate System* (IS-IS), and *Open Shortest Path First* (OSPF).

Network Meltdowns

Before beginning to work on a network so it will converge quickly, we need to set some realistic expectations. As mentioned previously, a routing protocol configured to react very quickly to changes in network topology tends to develop positive feedback loops, which result in a network that will not converge at all. Using the following example, consider how a single problem can produce feedback that causes a failure to cascade through the network.

Figure 1: Positive Feedback Loops in a Network



Suppose the link between routers D and G flaps, meaning that it cycles between the down and up states slow enough for a routing adjacency to be formed across the link, or for the new link to be advertised as part of the topology, but too quickly for the link to actually be used. In this situation, the adjacency (or neighbor relationship) between routers D and G forms and tears down as quickly as the routing protocol will allow.

While this is occurring, the routing information at routers E, F, and G is changing as quickly as the adjacency between D and G can form and tear down. This change in routing information is, in turn, passed on to C, which then must process it as fast as it possibly can. It is possible that the routing information presented to router C will overcome the ability of its processor to process the information, causing router C to fail, or drop its neighbor adjacencies.

At the same time, the constantly changing routing information at router B will also cause problems, possibly causing it to periodically drop its adjacencies, specifically with routers C and D. At this point, if the routers B, C, and D are all three consuming a large amount of memory and processing power adjusting to apparent topology changes because of changing adjacency states, the flapping link between routers D and G, which originally caused the problem, can be removed from the network, and the routing protocol will still not converge. This is what network engineers consider a classic *meltdown* in the routing system.

Solving the Meltdown

Typically, when a network engineer faces a network in this condition, the first step is to simply remove routing information from the system until the network "settles." This typically involves removing parallel (redundant) links from the view that the routing protocol has of the topology until the routing protocol converges. At this point, the network would be examined, routers reloaded as needed, and the parallel links brought back up. The network design might then be reviewed, in an attempt to prevent recurrence of a meltdown.

Routing protocol designers and developers would also like to move the point at which a routing protocol "melts" as far along the curve of network design as possible.

Of course, it is impossible to prevent all network meltdowns through protocol design; there are limits in any system where the implementation steps outside the "state machine," and the system will simply fail. But how would a routing protocol designer work around this sort of a problem in the protocol itself? The answer is actually very simple: Slow down.

The main problem here, from a protocol designer's point of view, is that routers D and G are simply reacting too fast to the changing topology. If they were to react more slowly, the network would not fall into this positive feedback loop, and the network would not melt. And, in fact, slowing down is really quite simple. Various methods of slowing down include:

- Not reporting all interface transitions from the physical layer up to the routing protocol. This is called *debouncing* the interface; most interface types wait some number of milliseconds before reporting a change in the interface state.
- Slow neighbor down timers. For instance, the amount of time a router waits without hearing from a given neighbor before declaring that a neighbor has failed is generally on the order of tens of seconds in most routing protocols. The dead timer does not impact down-neighbor detection on point-to-point links, because when the interface fails, the neighbor is assumed to be down, but there are other "slow-down" timers here, as well.
- Slow down the distribution of information about topology changes.
- Slow down the time within which the routing protocol reacts to information about topology changes.

All four of these methods are typically used in routing protocols design and implementation to provide stability within a routing system. For instance:

- In IS-IS, a timer regulates how often an intermediate system (router) may originate new routing information, and how often a router may run the *shortest path first* (SPF) algorithm used to calculate the best paths through the network.
- In OSPF, similar timers regulate the rate at which topology information can be transmitted, and how often the shorter path first algorithm may be run.
- In EIGRP, the simple rule: "no route may be advertised until it is installed in the local routing table" dampens the speed at which routing information is propagated through the network, and routing information is also paced when being transmitted through the network based on the bandwidth between two routers.

It seems like the simplest place to look when trying to decrease the time a routing protocol requires to converge, then, is at these sorts of timers. Reduce the amount of time an interface waits before reporting the transition to a down state, reduce the amount of time a router must wait before advertising topology information, etc. But when we consider implementing such changes, we remove much of the stability we have all come to expect in routing systems—the size a network can be built without melting down decreases below an acceptable threshold, even with modern processors, more memory, and implementation improvements in place.

There is another place to attack this problem: the frequency of changes within the network. This is the same concept—speed—from a different angle. How does looking at it from a different angle help us? By allowing us to see that it is not the speed of the network changes that causes the positive feedback loop, but rather how often the changes take place. If we could report the changes quickly when they occur slowly, and report them more slowly when they occur quickly, or if we could just not report some events at all, routing could converge much faster, and still provide the stability we expect.

The two options we want to examine, then, are not reporting every event, and slowing down as the network speeds up. First we will discuss these two options, and then discuss speeding up the reporting of network events, which plays a large role in decreasing convergence times.

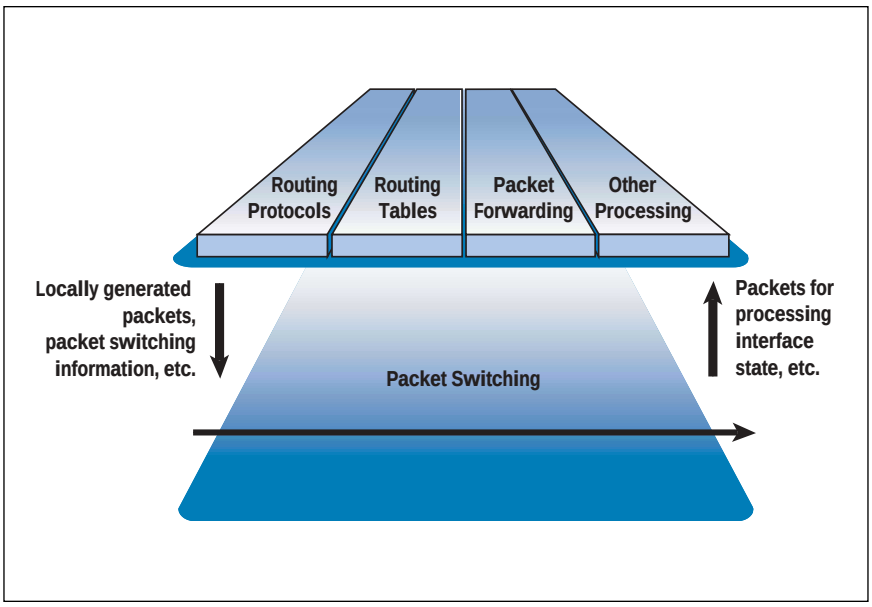
Do Not Report Everything You See (NSF and GR)

It sounds simple just to say that a router should not report every event within the network it is aware of, but it becomes more complicated as we consider the issues involved. What we need to do is sort out which events are important, in some sense, and which are not. For instance, if a router loses contact with an adjacent router because the adjacent router restarted for some reason, do not report the resulting change in topology until you are certain the neighbor is not coming back.

But the classic questions follow: How long do you wait before deciding the problem is real? And what happens to traffic you would normally forward to that neighbor while you are waiting? Finally, how do you reconnect in a way that allows the network to continue operating correctly? A technology recently incorporated in routing protocols called *Graceful Restart* (GR), combined with another technology called *Non-Stop Forwarding* (NSF), can combine to answer these questions.

Let's start at the bottom of the *Open Systems Interconnection* (OSI) model, at the physical and data link layers, and discuss the second question, what happens to traffic that would normally be forwarded while a router is restarting? Normally, this traffic would be dropped, and any applications impacted would need to retransmit lost data. How could we prevent this? We can take advantage of the separation between the control plane and the forwarding plane in a large number of modern routers.

Figure 2: Control and Data Plane Interaction in a Router



In some routers, such as the Cisco® 12000, 10000, 7600, and others, the actual switching, or forwarding, of packets is performed by different processors and physical circuitry than the control plane processes run on (such as routing protocol processes, routing table calculation, and other processes). Therefore, if the control plane fails or restarts for any reason, the data plane could continue forwarding traffic based on the last known good information.

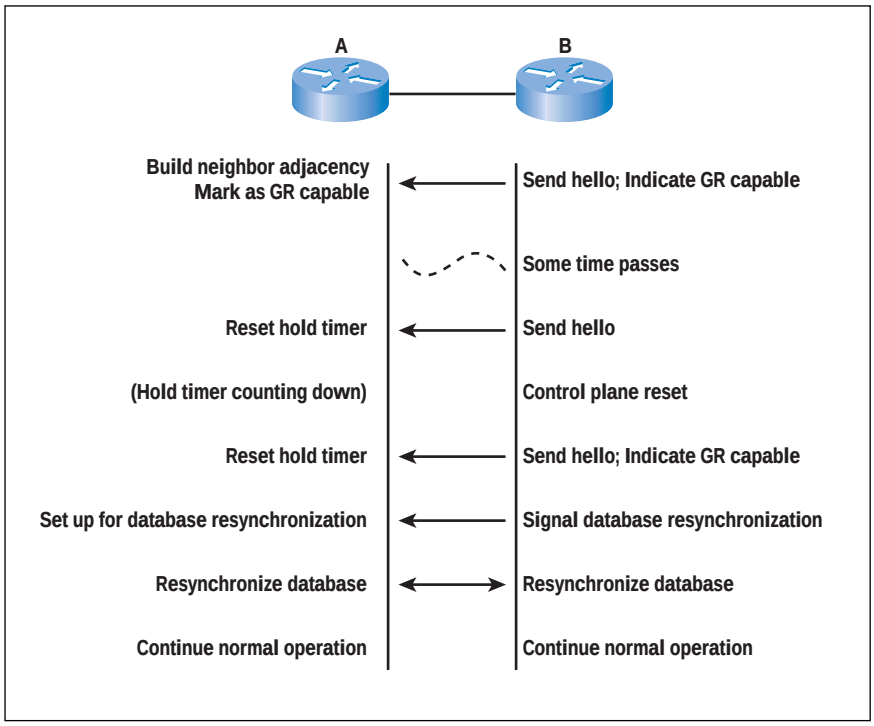
NSF, implemented through *Stateful Switchover* (SSO) and *Stateful Switchover+* (SSO+) in Cisco products, allows this continuous forwarding, regardless of the state of the control plane, to take place. Normally, when the control plane resets, it sends a signal to the data plane that it should clear its tables out, and reset, as well. With NSF enabled, this signal from the control plane simply acts as a signal to mark the current data as stale, and to begin aging the information out.

Now we need to be able to bring the control plane back up, resynchronize the routing protocol databases, and rebuild the routing table, all without disturbing the packets still being switched by the data plane on the router. This is accomplished through GR. GR starts by assuming two critical things:

- The normal hold times are acceptable, within this network environment, for reporting a network event or topology change. In other words, if a router's control plane fails, the event wouldn't be reported until the routing protocol's default hold or dead timers expire, whether or not GR is configured.
- The control plane on the router can reload and begin processing data within the hold or dead time of the routing protocol.

Let's examine how, in principle, GR works, so we can put these two requirements into context, and understand where GR is best deployed in a live network. Consider the following chart to understand how GR works between two peers of any generic routing protocol.

Figure 3: The Process of Graceful Restart



When two routers begin forming an adjacency (or neighbor relationship, or begin peering, depending on which routing protocol is being run between them), they exchange some form of signaling noting that they are capable of understanding GR signaling, and responding to it correctly.

[Note that this does not imply the router is GR-capable, only that it can support a neighboring router performing a GR. For instance, the Cisco 7200 supports switching modes only where the control and data planes are not cleanly separated, so it cannot fully support GR. It can, however, support the signaling necessary for a neighboring router to gracefully restart.]

Assume some time passes, and router B is transmitting Hello packets to router A normally, on a periodic basis. Each time router A receives one of these Hello (or *keepalive*) packets, it resets the hold, or dead, timer on router B, indicating that it should wait that amount of time before declaring router B down if it stops receiving Hellos. Now, at some point, after sending a Hello packet, the router B control plane resets. While the control plane is down, the router A hold timer is still counting down; the routing protocol does not reset the session. This is, in fact, normal routing protocol operation, which normally results in the packets forwarded by router A toward router B to be dropped. Because router B is NSF-capable, however, its data plane is still forwarding this traffic to the correct destination, even though the control plane is down.

If the router B control plane does not come back up within the dead or hold timer allowed by the routing protocol, router A declares the adjacency down, and begins routing around router B. This explains why the router B control plane must come back up within the hold interval of the routing protocol, one of the two assumptions we outlined GR as making at the beginning of this section. For this case, we assume that the router B control plane comes back up before the router A hold timer expires, and router B sends a Hello with no information other than indicating it is restarting.

When router A receives this Hello, it acts as though it has received a normal Hello, and simply keeps its adjacency with router B up. In other words, although router B may not know what the network it is connected to looks like at this point, router A does not report this failure to the rest of the network. Convergence time is, from a network standpoint, effectively reduced to 0.

When the router B control plane completes its reset, it then signals router A to begin resynchronizing their databases. The two routers then use some method specific to each protocol to resynchronize their databases, and begin operating normally, in a stable condition once again.

Slow Down When the Network Speeds Up

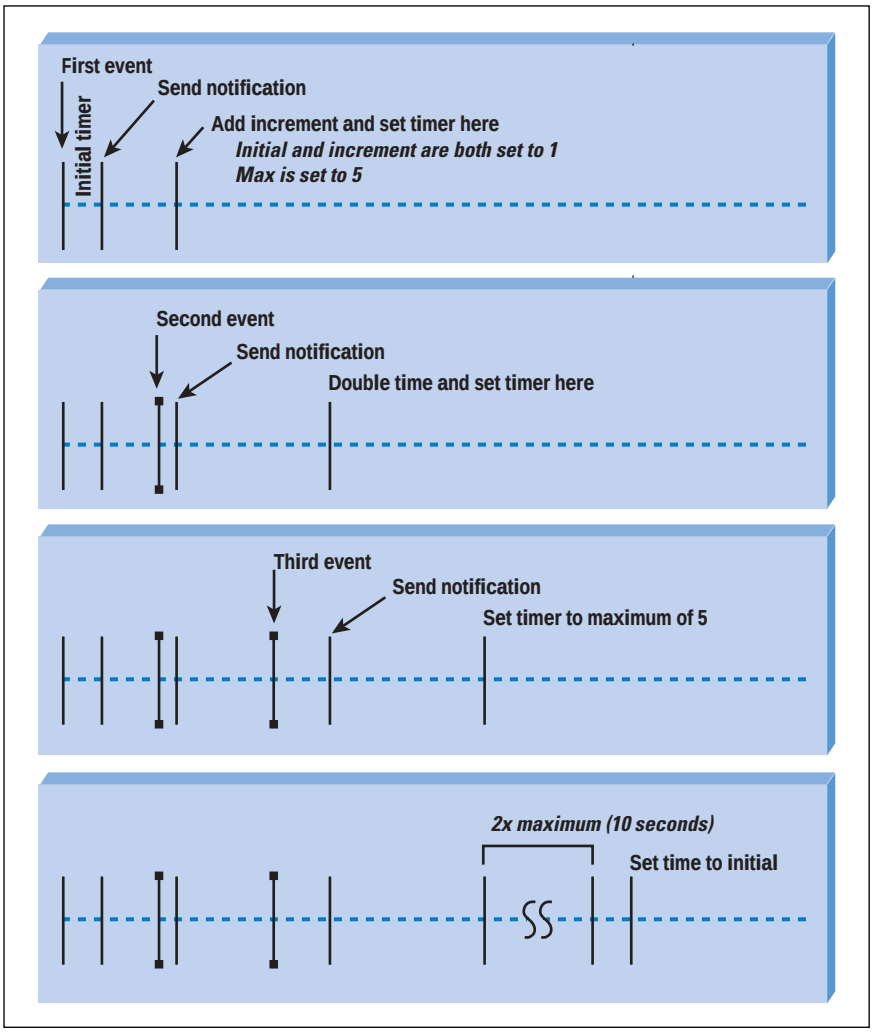
The second option we discussed originally was to attack the problem by reducing the frequency, rather than the number, of updates. What we want to do is to slow down the reporting of events when they occur more frequently (or when they occur rapidly), and speed up the reporting of events when they occur less frequently (or when they occur slowly). This is possible through a series of features built into Cisco IOS® Software within the last year or two, applying the concept of the *exponential timers.*

An exponential timer changes the amount of delay between an event occurring and the reporting of that event by the frequency at which the event occurs—possibly not reporting the event at all, in some situations. Two implementations of exponential timers are *exponential backoffs* and *dampening.* Let's examine each of these individually, and then consider where they are implemented in Cisco IOS Software.

Exponential Backoffs

Consider the following figure to examine how exponential backoff works.

Figure 4: Exponential Backoff



When the first event occurs, a timer is set to the initial time, 1 second in this case, meaning that the router waits for one second before notifying other routers in the network about the event. When the notification is sent, the router adds the initial timer to the increment, and sets a timer for this period. We call this timer the *backoff timer*.

When the second event occurs, the backoff timer is still running; the router waits until this timer expires to send the notification about this event occurring. When this notification is sent, the backoff timer is set to twice the previous setting or the maximum backoff time, whichever one is shorter. In this case, doubling the backoff timer results in 4 seconds, so it is set to 4 seconds.

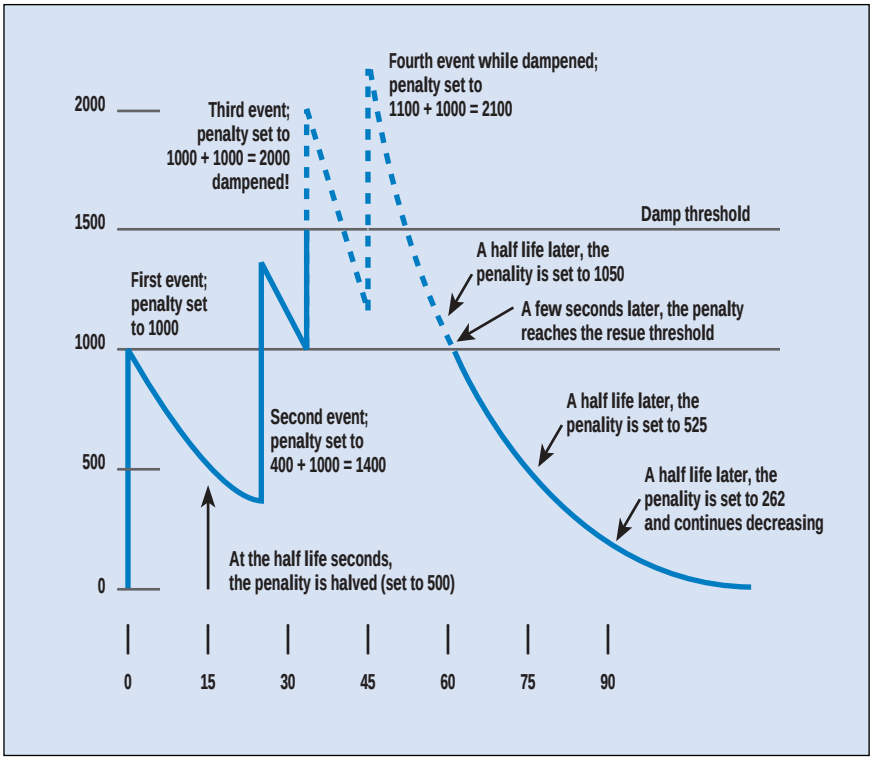
When the third event occurs, the backoff timer is still running; the router waits until the timer expires before sending any notification of the event occurring. Again, the timer is doubled, this time to 8 seconds, and compared to the maximum time, which is 5 seconds. The shorter of the two times is taken, so the backoff timer is now set for 5 seconds.

At this point, any future events will be reported only at 5-second intervals, as long as one event occurs at least every 5 seconds. If no events occur for an interval of 10 seconds, the timers are all reset to their initial condition, so the initial timer is set to 1 second, and the backoff timer is not set at all.

Dampening

Dampening, or damping, is also an exponential backoff mechanism similar to the exponential backoff algorithm we examined previously. The primary difference is that dampening is applied to events that have a Boolean component; a route that is either advertised or withdrawn, an interface that is either up or down, etc. Exponential backoff simply deals with events in general, whereas dampening adds value based on the type of event, as well as the frequency at which the event occurs. Consider the following figure to understand dampening.

Figure 5: Dampening Over Time



In dampening, the desirability of reporting an event is set using the *penalty*; the higher the penalty applied to a given item, such as a route or an interface, the less desirable it is to advertise changes in the state of that item. Dampening always leaves the item in the "off," or "down," state, when it stops reporting state changes; this is called the *dampened* state. A penalty is normally added when transitioning from "down" to "up" in most dampening systems.

Here, we start at time 0, with a penalty of 0; when the first event occurs, a penalty of 1000 is added, making the total penalty 1000. As time passes without another event occurring, the penalty is decreased, based on the *half life*. Each time the half life passes, in this case 15 seconds, the current penalty is halved, so after 15 seconds, the half life is set to 500.

A few seconds later, while the penalty is still decreasing, the second event occurs; 1000 is added to the current penalty, making the total penalty 1400. Again, as time passes, the penalty decays exponentially, reaching 1000 before the third event occurs. When the third event occurs, 1000 is again added to the total penalty, so it reaches 2000—which is above the *damp threshold*, so future events are dampened by simply leaving the interface or route in the down state.

Again, as time passes, the penalty is cut in half for each passing half life, reaching 1100 before the fourth event occurs. When the fourth event occurs, 1000 is again added, making the penalty 2100, and leaving us in the dampened state until the penalty can be reduced again. Over time, the penalty finally drops to 1000 (at around 60 seconds in the example), which is the *reuse threshold*. At this point, state changes in the item being tracked are once again reported as they occur, unless the penalty reaches the dampening threshold at some future point.

So, dampening reacts to events by simply not reporting events if they occur too frequently, whereas exponential backoff reacts to events by reporting each event that occurs, but slowing down the reporting of events as they occur more frequently.

Speeding Up the Reporting of Events

When we have some methods in place to prevent a network meltdown when events occur, we can consider ways to discover events faster. Primarily, these techniques are used in conjunction with exponential backoff and dampening.

There are two ways to detect a down neighbor or link: *polling* and *event driven*. We will briefly discuss each of these, and some various techniques available in both cases.

Polling

One method commonly used for detecting a link or adjacency failure is polling, or periodically sending Hello packets to the adjacent device, and expecting a periodic Hello packet in return. The speed at which Hello packets are transmitted and the number of Hello packets missed before declaring a link or adjacency as failed are the two determining factors in the speed at which polling can discover a failed link or device.

Normally, a neighbor or link is declared down if three Hello packets are lost, meaning that the hold time, or the dead time, will always be about three times the Hello time, or polling interval. Normally, for Layer 2 links and routing protocols, the Hello interval is measured in seconds. For instance:

- EIGRP running over a point-to-point link sends one Hello every 5 seconds, and declares a neighbor down if no Hellos are heard for 15 seconds.
- EIGRP running over a lower-speed point-to-multipoint link sends one Hello every 60 seconds, and declares a neighbor down if no Hellos are received in 180 seconds.

- OSPF normally sends a Hello every 10 seconds, and declares a neighbor down if no Hellos are heard for 40 seconds.
- *Frame Relay Local Management Interface* (LMI) messages, the equivalent of a Hello, are transmitted every 10 seconds. If an LMI is not received in 30 seconds, the circuit is assumed to have failed.
- *High-Level Data Link Control* (HDLC) keepalive messages are transmitted every 10 seconds. If a keepalive message is not received within 30 seconds, the circuit is assumed to have failed.

Fast Hellos can decrease these timers to Hello intervals on the order of 300 milliseconds, and dead timers of around 1 second.

The primary problem with fast Hellos is scaling, particularly in receiving and processing fast Hellos from a large number of neighboring routers. For instance, if a router has 1000 neighbors and is using a Hello interval of 330 milliseconds, the router has to be able to receive and process 3000 Hellos per second and send 1000 Hellos per second. Timers in this range leave little room for processes that consume a router processor for long periods of time, short-term packet loss on a link due to congestion, and other factors.

Event Driven

Rather than polling at a fast interval, event-driven notifications rely on devices within the network that can sense the state of a link through lower layers (electrical, electronic, or optical state) to notify the routers attached to the link when the link has failed. SONET links are probably the best example of media with built-in capabilities for sensing link failures and notifying attached devices. This Tech Note on Cisco Online:

`http://www.cisco.com/en/US/tech/tk482/tk607/technologies_tech_note09186a0080094522.shtml`

... provides information about SONET alarms. There are also techniques that can be used to speed up the reporting of failed links in Frame Relay circuits, and techniques are being developed for allowing switches to notify devices attached to an Ethernet VLAN about a loss of connection to an attached device.

Implementations

Now that we have discussed what exponential backoff and dampening are, we can consider how they are implemented, and how their implementation helps you build highly available networks (through fast convergence) without risking major network instability along the way. We start by examining where dampening is implemented, and then follow that with a discussion about where exponential backoff is implemented. These sections do not provide a great deal of detail on the implementation of these features; vendor documentation and other sources of information (such as the forthcoming book *Designing to Scale*) should be consulted for technical details.

Dampening

Dampening is currently implemented in two places:

- *Border Gateway Protocol* (BGP) route flap dampening
- Interface dampening

BGP route flap dampening is a well-known technology, deployed in the Internet on a wide scale to increase the stability of the Internet routing table.

Interface dampening allows the network engineer to prevent rapidly flapping interfaces from having a wide-ranging impact on the entire network. When an interface fails and comes back up numerous times within a short time period, the interface is placed in the down state from an IP perspective, and not advertised within routing protocols, or used for forwarding packets.

It is important to note that the interface is allowed to change states freely at Layer 2; an interface that continues to change state rapidly continues to accumulate penalties, and continues to show down to the IP subsystem.

Exponential Backoff

Exponential backoff is implemented in several places in link state protocols at this point, including:

- The ability to exponentially back off the amount of time between a change in the network topology being detected and the transmission of a link state packet being transmitted to report the change; exponential backoff has been applied to the link state generation timer.
- The ability to exponentially back off the amount of time between receiving a link state packet reporting a change in the network topology, and running SPF to recalculate the path to each reachable destination in the network; exponential backoff has been applied to the SPF timer.

Fast Hellos

Each routing protocol has a different limit on how Fast Hellos can be transmitted and how often they must be received for a neighbor to be considered alive. OSPF and IS-IS have both implemented the fastest Hellos, with a minimum of 330 millisecond Hellos, and a dead interval of 1 second.

EIGRP can run with Hellos as fast as one per second, with a 3-second dead time. BGP can use similar timers, with a keepalive interval of 1 second.

Caution should be used when configuring Fast Hellos on a network. Congestion, high processor use, and other problems can cause false down indications that may cause higher rates of network failure than would normally occur.

Deploying GR and Fast Convergence Technologies

We now have a full range of options we can use to improve network availability, including GR and NSF, event dampening, and fast convergence techniques. How can we deploy these in a real network to improve network uptime? Generally, the technologies can be placed in one of three categories:

- *Fast reaction to node or link failure, to route around the failure.* We use Layer 2 techniques and Fast Hellos to quickly determine when an adjacent node, or a link to that node, has failed.
- *Slow reaction to node of link failure, combined with routing through the failure.* We rely on moderate speed reactions to node failures to allow resynchronization of routing data while forwarding of traffic continues.
- *Fast recalculation of the best path when a topology change has been reported.*

As we can see, the first two are complementary; we could not deploy both of them in the same place in the network. The third one, fast recalculation, can be deployed with either (or both) fast reaction and slow reaction techniques to increase network availability. The primary question then becomes: which of these two techniques do you deploy in your network, and where?

The basic premise behind making this design decision follows:

- If there is a fast, online backup available to a node or a link, it probably makes more sense to route around any problems that occur as rapidly as possible.
- If any existing backup is going to take a good deal of time to bring online, or there is no backup path (such as a single homed remote office, or a remote office with dial backup), it probably makes more sense to route through any problems.

In general, then, we want to deploy techniques that improve network convergence time everywhere—techniques that bring down the time a network is down when a failure occurs, is detected, and a new path calculated. At the same time, we want to evaluate each point in the network we would like to protect from failure, and determine the best means to protect that point of failure: redundancy with fast down detection, GR, or NSF.

Fast, stable networks are possible with today's techniques in routing; some large networks, with several hundred routers, measure their convergence times in the milliseconds, with 1 second as their outside convergence goal.

RUSS WHITE is on the Cisco Systems Routing DNA Team in Research Triangle Park, North Carolina, specializing in the deployment and architecture of routing protocols. He has coauthored books on routing protocols, network design, and router architecture, regularly speaks at the Cisco Networkers conference, and is active in the Internet Engineering Task Force. Russ can be reached at **riw@cisco.com**. This article offers a high-level overview of material covered in depth in a forthcoming network design book, *Designing to Scale*, being published through Cisco Press.

The Lures of Biometrics

by Edgar Danielyan, Danielyan Consulting LLP

This article introduces biometrics and discusses some of the complex issues associated with use of biometrics for identification and authentication of individuals and its impact on both standalone and networked information systems, as well as on physical security. The agenda is not to show whether biometrics is your best investment or a useless thing—these two polar viewpoints share the same quality of being oversimplifications, to say the least. It also certainly does not purport or try to tell everything there is to tell about biometrics or its applications. Legal and social implications of biometrics are also not discussed in this article because these would differ considerably, depending on the legislation and cultural traditions of countries concerned; we also do not consider the complex performance, design, and implementation questions, because these are of too specialized nature—for more in-depth coverage of these topics a list of biometrics organizations and publications are provided at the end of this article, along with a list of references.

Before we continue, it would be useful to examine the current deployment of biometrics outside testing laboratories and the corporate perimeter. With the U.S. government fingerprinting and taking photographs of some of the visitors coming to the United States beginning January 5, 2004, under the US-VISIT program, biometrics and associated issues such as privacy and personal data protection are bound to get unprecedented levels of publicity[1]. Although it is too early to judge whether this innovation will actually contribute to overall security of the country or rather increase the general confusion surrounding security procedures, it has already resulted in more questions asked than answered. To some of its proponents, biometrics is a magic technology that would contribute to the security of their societies, to others the same technology heralds the coming of a police state and erosion of personal privacy and liberties and discrimination against (potentially not only) foreign citizens. Indeed, that was the opinion of Julier Sebastiao da Silva, a federal judge in Mato Grosso state of Brazil, who ordered similar measures to be taken in the case of U.S. citizens visiting Brazil[2]. Despite the announcement of Brazil's federal police that they may well seek to have this judgment overturned, this is a significant event because it illustrates that the use of biometrics is not only a technical procedure but also has its far-reaching social, legal, and international implications. It is immaterial whether this judgment will be upheld or overruled—it is the fact that introduction of the mandatory use of biometrics at borders resulted in such a response that is important.

Earlier announcement by the U.S. authorities that they expect the visa-waiver countries whose citizens currently may enter the U.S. without visas, simply upon presentation of their passports, to provide biometric data in newly issued passports also resulted in different reactions, ranging from support for the measure to outright condemnation[3].

Aside from the huge technological and logistical work that must be done in order to introduce biometrics into passports, these requirements also pose considerable legal and social issues in countries with strong personal privacy and data protection legislation in place. However, one thing is clear—biometrics ceases to be an exotic and little-used technology and is bound to be increasingly used in one way or another.

This article is organized as follows. First biometrics and related concepts are introduced, along with descriptions of the most widely used and understood physiological and behavioral biometrics. We will also see how biometric systems fail when inadequately designed or implemented. Later we describe the system and design issues of biometrics, such as security, accuracy, speed, resilience, privacy, and cost of biometric identification and verification systems, as well as practical applications of biometrics in network authentication and international travel documents.

Definition of Biometrics

A *biometric* is a physiological or behavioral characteristic of a human being that can distinguish one person from another and that theoretically can be used for identification or verification of identity. For a biometric to be practically useful, ideally it should be unique, universal, permanent, recordable, and acceptable—more on these properties of practical biometrics later.

Authentication in General

Authentication is the second step in the identify-authenticate-authorize process, which is done countless times every day by humans and computers alike. When speaking about human authentication, basically we have three choices: using something we know (such as passwords and passphrases), something we have (such as access tokens, smart cards, and so on) or something we are (biometrics). There is no "best" authentication method; each has its pros and cons, depending on the application, the users, and the environment. Whatever authentication method we use, we can make it stronger by using one or both of the other methods. An example of strong authentication would be a system that requires possession of a smart card, knowledge of a password or *Personal Identification Number* (PIN), and biometric verification. Obviously to steal or fake all three would be much more difficult than to steal or fake any one of these—however, more expensive and laborious to operate as well. The other two factors—the time of access and the location of subject—may also be used for access control, but usually only as auxiliary factors.

What You Know

Unquestionably the most widely used method of authentication, passwords, passphrases, and PINs share both pros and cons with each other. Moreover, an advantage in one situation easily becomes a problem in another—an example being the ease of password sharing. Passwords are easy to change, but are also easy to intercept. Systems can force the use of strong passwords, but the user may respond by storing or transmitting them in such a way that the added security is effectively reduced to nil.

Unauthorized disclosure of a password is not usually detected until after unauthorized access has already taken place. Passwords are also vulnerable to guessing, dictionary, and brute-force attacks. On the other hand, they require no additional hardware, they are an accepted method of authentication, and they are well-understood—even by the most technologically challenged part of human species.

What You Have

Smart cards, access tokens (both challenge-response and time-based), and other "what you have" authentication methods solve some of the problems associated with "what you know" authentication, but they create a set of different problems. Unlike theft of a password, theft of a smart card or access token can, of course, be easily detected. Unlike passwords, smart cards usually cannot be used simultaneously by two or more parties in different places. However, "what you have" authentication devices may be lost, damaged, and stolen. They may also run out of power (if self-powered) or may be prone to power-, synchronization- and time-based attacks if externally powered. They may also be subjected to reverse engineering and other treatment, which may compromise their security.

What You Are: Biometric Authentication

There are two biometric authentication methods: biometric verification and biometric identification of identity. Biometric identification is also sometimes referred to as *pure biometrics* because it is based only on biometric data and is more difficult to design and operate—but alas, pure biometrics is not the most secure, useful, or efficient one. Also, both methods can not always be used with all biometrics—some biometrics can only be used in verification mode because of their intrinsic properties.

Verification

Biometric verification uses entity IDs and a biometric—in this case biometric merely serves to prove identity already declared by the entity—which may be done using something you know (a username) or something you have (a smart card). Biometric (something you are) works to actually complete the authentication process. Hence, the biometric database keeps a list of valid entity IDs (which may be said to serve as primary keys to the database) and corresponding biometric templates, and compares ("matches") the stored template with the biometric provided. The result of this comparison is either an accept or reject decision based on a complex algorithm and system settings (refer to the section "Matching").

Identification

Unlike biometric verification of identity, biometric *identification* is based solely on biometrics. The biometric serves as both the identifier and the authenticator. The biometric database contains the enrolled biometric templates, and they all are compared against the provided biometric to find a match. Biometric identification may be described as "putting all your eggs in one basket," partly because somehow faking or stealing a biometric compromises both the ID and the authenticator.

A biometric identification system may operate in one of the two modes: positive identification or negative identification. In a positive identification biometric system, the provided biometric must be in the database and there must be only one match to positively identify the person. The risks present in a biometric system are false acceptance and false rejection, whereas unauthorized subjects are incorrectly accepted, or authorized ones are denied identification, resulting in a denial of service. A negative identification system, in contrast, works by determining whether the provided biometric is not in the database.

Enrollment

Regardless of the type of a biometric system, *enrollment* is a mandatory part of the process. Biometric enrollment is the registration of subjects' biometrics in a biometric database. Positive enrollment results in a database of recognized persons' biometric templates that may be later used for positive identification or verification. Negative enrollment results in a database of "excluded" persons, a black list if you wish. Security and reliability of the enrollment process and the biometric database are fundamental to the security of the entire system, but in practice they are difficult to achieve because of the myriad of issues that affect collection, transmission, storage, and usage of biometric data (see "Security" and "Privacy," later in this article for an overview of just some of the risks).

Matching

After an individual is enrolled—that is, the individual's biometrics are scanned and registered in the biometric database—*matching* is the next step. Biometric matching is essentially the comparison of the enrolled person's known biometric data stored in the biometric database in the form of biometric templates—binary representation of biometric sample—with the biometric provided by the individual at the identification or verification time. However, biometric matching is a pattern-recognition problem and not a simple bit-by-bit comparison—representation of the same biometric taken by two input sensors or taken at two different points in time does not match bit by bit because of numerous factors such as sensor resolution, system noise, and so on. Therefore, a degree of likeness (usually referred to as the *matching score*) is used to express how like the stored biometric is to the provided biometric. A *threshold level* is used to decide whether the matching score is high enough to be considered a match—if the score is at or below the threshold level, matching fails. This threshold level is one of the many variables that affect the accuracy—and hence security—of biometric authentication systems.

For biometric identification applications, the provided biometric is compared against all entries in the database and should result in only one successful match to result in positive identification. In biometric verification systems, the provided biometric is compared only with the biometric template or templates corresponding to the specified identity. As a result of biometric matching, the following system errors may occur:

- *False match or acceptance:* This occurs when the system decides that the two biometrics (the one stored in the database and the one provided now) are the same, when in reality they are not. The rate of false matches is known as *False Matching Rate* (FMR) or *False Acceptance Rate* (FAR). False acceptance is a confidentiality and integrity risk.
- *False nonmatch or rejection:* This is expressed as *False Rejection Rate* (FRR), and *False Nonmatching Rate* (FNMR). False nonmatch is when the system erroneously decides that biometrics are from different identities while in reality they are from the same person. False rejection is an availability risk.

In practice, both FRR and FAR do not equal zero, and in different applications one of them may be more important than the other. In an application that requires higher security (and hence as low FAR as possible), users may be troubled with high false rejection rates; whereas in an application that can accept somewhat higher false acceptance rates (such as public transport), false rejection rate is of more concern because of convenience and manual processing concerns. When FAR and FRR meet, that is the *Cross-over Error Rate* (CER). The lower the CER, the better—hence it is frequently used to express accuracy of biometric systems (although it is not the infallible measure as some suppose). Additionally, *Failure to Acquire* (FTA) errors occur when an individual does not have the required biometric or the biometric cannot be read by the sensor; and *Failure to Enroll* (FTE) is when a part of the targeted population may not be enrolled for whatever reason (such as a FTA). These errors directly affect the practicality of biometrics and must be accounted for with regard to the projected population of users.

Practicality of Biometrics

Writing in the December 1994 issue of *Information Technology & People* ("Human identification in Information Systems: Management Challenges and Public Policy Issues")[4] ten years ago, Roger Clarke proposed some criteria that should be met in order for a biometric to be practically usable:

- *Universality:* Every relevant person should have an identifier.
- *Uniqueness:* Each relevant person should have only one identifier, and no two people should have the same identifier.
- *Permanence:* The identifier should not change, nor should it be changeable.

- *Indispensability:* The identifier should be one or more natural characteristics, which each person has and retains.
- *Collectibility:* The identifier should be collectible by anyone on any occasion.
- *Storability:* The identifier should be storable in manual and in automated systems.
- *Exclusivity:* No other form of identification should be necessary or used.
- *Precision:* Every identifier should be sufficiently different from every other identifier that mistakes are unlikely.
- *Simplicity:* Recording and transmission should be easy and not error-prone.
- *Cost:* Measuring and storing the identifier should not be unduly costly.
- *Convenience:* Measuring and storing the identifier should not be unduly inconvenient or time-consuming.
- *Acceptability:* Its use should conform to contemporary social standards.

Although some of these criteria may be argued over, this set is nevertheless a useful reference. An interesting point is that no known biometric completely satisfies all of these criteria, perhaps proving that these are not strict "must haves" but instead guidelines to be accounted for.

Types of Biometrics

Two broad categories of biometrics exist: *physiological* biometrics (such as fingerprints, hand geometry, iris recognition) and *behavioral* biometrics (such as signature and voice biometrics). Physiological biometrics is based on direct measurements and data derived from measurements of a part of the human body, whereas behavioral biometrics is based on measurements and data derived from human actions, and indirectly measures characteristics of the human body over a period of time.

Physiological Biometrics

Relatively widely understood and used physiological biometrics are fingerprint recognition, face recognition, hand geometry, and iris recognition. These methods are introduced in the following sections.

Fingerprint Recognition

It is believed that no two persons share the same fingerprints—not even identical twins—because the fingerprint patterns are part of a person's phenotype and do not apparently depend on genetics[5]. Fingerprints have been used to identify humans for a long time—there is some evidence that thousands of years ago ancient Chinese were aware of the uniqueness of fingerprints[6], not speaking about their current use in forensic science and law enforcement. The traditional fingerprint acquisition mechanism—finger into ink and then on to paper—obviously is not usable in many—if not most—noncriminal applications.

Currently there are four known inkless fingerprint acquisition mechanisms considered suitable for use in practical biometrics.

Optical Sensing

Optical fingerprint sensing works by acquiring light reflected from the finger surface through a special prism. The result is an image of the finger surface. The downside of this method is that wet, dirty, or dry finger skin may result in a bad image.[7]

Thermal Sensing

With the thermal sensing method, a thermogram of the finger surface is taken and the resulting image is used.[8]

Capacitance Sensing

Because of differing capacitance of the ridges and valleys of fingers, a *Complementary Metal-Oxide Semiconductor* (CMOS) capacitance sensor can obtain an image of the finger when it is touched. However, like optical sensing, cpacitance sensing may be negatively affected by dry, dirty, or wet skin. [9]

Ultrasound Sensing

Ultrasound sensing works by using an ultrasound beam to scan the skin surface. Ultrasound sensing is not affected much by dry, dirty, or wet skin but takes longer to perform and the ultrasound sensing equipment is usually not compact and consequently not widespread.[10]

In addition to the mentioned issues of wet, dry, or dirty skin, numerous other factors may also affect the quality or the very possibility of taking a fingerprint. For example, although the absolute majority of people have at least one finger, many people may also have damaged skin or skin illnesses that may degrade the quality of fingerprints or render them unusable. Fingerprint matching approaches may be broadly categorized into three classes: feature techniques, imaging techniques, and hybrids of the two. In feature-based fingerprint matching techniques, a symbolic representation of the fingerprint, defined by so-called *minutiae*, is created from the fingerprint image, and it is this representation that is later stored and used to match fingerprints—not the raw fingerprint image itself[11]. Imaging techniques use the fingerprint images directly—image correlation algorithms are then used to compare the fingerprints[12].

The Mighty Fingers

If the defending technology is expensive and complex, it does not mean the attacking technology will also be complex and expensive—this has been proven by many successful security attacks. Tsutomu Matsumoto of the Yokohama National University successfully fooled numerous fingerprint readers into accepting fake fingers made of gelatin with a 80-percent success rate, sending a shock wave among biometrics proponents[13].

In a paper ambiguously entitled "Impact of Artificial Gummy Fingers on Fingerprint Systems," co-authored with H. Matsumoto, K. Yamada, and S. Hoshino and presented at the Optical Security and Counterfeit Deterrence Techniques IV conferene (Proceedings of the *International Society for Optical Engineering*, 2002), Matsumoto describes relatively easy ways to create artificial clones of fingers using cheap and freely available materials such as gelatin, free molding plastic, and photosensitive printed circuit boards.

Not only was he able to create a copy of a live finger that was good enough to fool most fingerprint readers used in the experiment, he also created an artificial finger using a latent fingerprint left on a glass, which was also accepted as genuine. In addition, Matsumoto mentions several other attack vectors against fingerprint systems, including instances where the registered finger is presented by an armed criminal, under duress, or on a sleeping drug; a severed fingertip of the registered finger; or a genetic clone of the registered finger.

Even if we disregard the last possibility as too expensive and unlikely, the others are indeed very real and must be disturbing to current users of fingerprint-based identification or verification systems. After this research was published, Bruce Schneier wrote in the May 2002 issue of his monthly newsletter CRYPTO-GRAM[14]:

> "There's both a specific and a general moral to take away from this result. Matsumoto is not a professional fake-finger scientist; he's a mathematician. He didn't use expensive equipment or a specialized laboratory. He used $10 of ingredients you could buy, and whipped up his gummy fingers in the equivalent of a home kitchen. And he defeated eleven different commercial fingerprint readers, with both optical and capacitive sensors, and some with "live finger detection" features. (Moistening the gummy finger helps defeat sensors that measure moisture or electrical resistance; it takes some practice to get it right.) If he could do this, then any semi-professional can almost certainly do much much more. More generally, be very careful before believing claims from security companies. All the fingerprint companies have claimed for years that this kind of thing is impossible. When they read Matsumoto's results, they're going to claim that they don't really work, or that they don't apply to them, or that they've fixed the problem. Think twice before believing them. "

Face Recognition

One of the most powerful drivers behind the use of face recognition is the fact that we all use face recognition every day to recognize people—so it seems to be one of the most acceptable biometrics we have (unlike, for example, fingerprints, which are often associated with criminal prosecution), not speaking about photographs that have been used for identification for many years[15]. However, despite progress in this area of biometrics, face recognition is still not accurate and dependable enough, and factors such as aging, changing hairstyles, beards, and moustaches only make reliable face recognition more difficult. Bruce Schneier, in his recent book *Beyond Fear*, had the following to say about the usefulness of face recognition systems[16]:

> "I'll start by creating a wildly optimistic example of the system. Assume that some hypothetical face-scanning software is magically effective (much better than is possible today)—99.9% accurate. That is, if someone is a terrorist, there is a 1-in-1000 chance that the software fails to indicate "terrorist" and if someone is not a terrorist, there is a 1-in-1000 chance that the software falsely indicates "terrorist." In other words, the defensive-failure rate and the usage-failure rate are both 0.1%. Assume additionally that 1 in 10 million stadium attendees, on average, is a known terrorist (this system won't catch any unknown terrorists who are not in the photo database). Despite the high (99.9%) level of accuracy, because of the very small percentage of terrorists in the general population of stadium attendees, the hypothetical system will generate 10,000 false alarms for every one real terrorist. This would translate to 75 false alarms per Tampa Bay football game and one real terrorist every 133 or so games."

Of course these issues do not apply exclusively to face recognition systems, but we get the idea—a system that generates so many false alarms and catches so few terrorists is not going to be successful. This was proven on several occasions. First at the Palm Beach International Airport, where a face recognition system failed by providing less than 50-percent recognition rate and generating a large number of false positives, resulting in a decision by the airport not to use the system at all[17]. Almost the same happened in the second case, at a face recognition system trial at the Boston Logan International Airport[18].

Hand Geometry

Features measured and used by hand geometry biometrics typically include length and width of fingers, different aspect ratios of palm and fingers, thickness and width of the palm, and so on[19]. Existing hand geometry systems mostly use images of the hand. Like face recognition, hand geometry is a user-friendly technology that scores higher on the acceptability test than, for example, fingerprints. It is also relatively more easily measurable and recordable than some other biometrics. Several patents have been issued for hand geometry systems, but there is not as much research as on fingerprints[20]. However, because of its biometric properties, hand geometry is not suitable for use in the identification mode.

Iris Recognition

Iris recognition-based biometric systems are believed to be very reliable and accurate[21]. Like fingerprints, the iris image is a part of human phenotype and is believed to be unique in every individual. Perhaps one of the most known cases of deployment of the iris recognition system is the Privium at Amsterdam's Schiphol International Airport. Frequent travelers may enroll in the system to enjoy fast border crossing by simply looking at the iris scanner, which authenticates the person and opens the gate[22]. In February 2004, an iris recognition system will also be piloted at the Frankfurt International Airport, and if the six-months-long trial concludes successfully, the system may be installed and deployed in 18 European countries[33]. Obviously, iris recognition would not work for people who are missing both eyes or who have serious eye illnesses that affect the iris.

Behavioral Biometrics

Two of the most used behavioral biometrics are signature- and voice-based systems. Another behavioral biometric, keystrokes (where the timing between successive key pressings is used), seems to receive increasing attention and use.

Signature

In use for centuries, signatures enjoy a high degree of acceptance, largely because of their everyday use and familiarity, but as a behavioral biometric, signatures lack permanence: they may change at the will of a person, or under influence from such factors as illness, mental state, medicines, emotions, or age. For these and other reasons, signature-based biometric systems function in the verification and not in the identification mode.

Two subtypes of signature verification systems exist: static signature verification systems, where only the graphical representation (image) of the signature is used, and dynamic signatures, where the dynamics, pressure, and speed of the movement of a special pen are used for verification. Although the first method does not require any special hardware, the dynamic signature verification requires the use of special electronic signature readers or high-quality tablets. It is understood that dynamic signature verification is more secure and reliable than static signatures[23]. However, some people do not have consistent signatures, resulting in increased false rejection rates to unacceptable levels and severely affecting the practical use of signature-based biometric systems.

Voice

Voice recognition systems (not to be confused with speech recognition systems, which are concerned with the actual words said and not the identity of the speaker) depend on numerous characteristics of a human voice to identify the speaker. Voice recognition holds much potential because it is acceptable and it does not require expensive input devices, unlike some other biometrics. Like face recognition, voice recognition is something we humans do many times a day; additionally, voice recognition is ideal for many practical and widespread telephony applications, and in theory voice recognition systems may even function in the background without forcing the users to go through a separate identification and verification process, saving us from another password to remember. But as usual, voice recognition systems also have their fair share of potential problems. As we all know, some people with exceptional vocal abilities may skillfully imitate others' voices, potentially defying such systems. Another issue is the ease of sound recording and replay, so any voice recognition system must be designed to withstand "record and replay" attacks.

Voice recognition also is influenced by the usual suspects—illness, mental state, emotions, age—which may substantially modify an enrolled subject's voice to a degree that it does not match the stored templates anymore. Several voice recognition models varying in accuracy and complexity exist.

The *fixed-text* model involves a person saying a word or phrase previously recorded and enrolled in the biometric database. The verification process is the simple comparison, possibly accounting for some allowable differences. However, if this word or phrase can be recorded, the entire system fails, because it is fairly easy to reproduce words and phrases.

Another model is *text-dependent*, meaning the system instructs the person to speak words or phrases—naturally this system is less prone to replay attacks because supposedly the person does not know in advance what words or phrases the system will ask for. A hybrid system, also known as *conversational voice verification*, combines something you are—your voice—and something you know—such as a password—to provide a higher degree of verification accuracy and reliability, and this system may well be the best choice in practice[24], so multimodal biometrics may hold the key to more accurate and practical biometric authentication. Again, we should keep in mind that some people cannot use this biometric for one reason or another.

System and Design Issues

The following is a quick overview of only some of the most important biometric system design and implementation considerations:

Security

Biometrics is invariably associated with security, hence the biometric system itself should be reasonably secure and trustworthy. Not only should the system provide the required functionality, but we also should have a degree of security assurance. Keeping in mind our track record of creating secure complex systems (almost an oxymoron), we should not really have high expectations this time either. If we have learned a lesson, it is that systems fail and malfunction, so recovery and compensating mechanisms should be in place from the beginning, and even the most sophisticated system should be expected to fail sooner or later, one way or another. Some of the biometrics security issues are discussed in the following section.

Rogue Sensors and Unauthorized Acquisition (theft) of Biometric Samples

One of the risks associated with the use of biometrics for identification or verification is that a biometric cannot be changed by definition—your fingerprint is your fingerprint and there is no easy way to change it—so if it is stolen and used to create a fake finger to impersonate you, there is not much you can do about yours. Therefore, the issue of mutual authentication of the individual and the sensor is of much importance. In practice, however, as illustrated by numerous stories about rogue *Automated Teller Machines* (ATM) harvesting unsuspecting victims' card and PINs, this would prove to be a difficult task. Unlike, for example, smart cards, which may use cryptographic protocols to establish with whom they are communicating, we humans have no secure way to ascertain whether the biometric reader attached to a computer somewhere is indeed under control of (let's say) a genuine Internet banking application and will not relay or store our biometric template without authorization.

In contrast, bank customers asked to authenticate themselves at a bank counter may have a reasonable expectation that their biometric will be used by the same bank for lawful purposes only—because of their and the sensor's physical location (so called location-based authentication). Still, unauthorized acquisition and use of biometrics remains one of the issues to be considered in any practical implementation.

The fact that not all biometrics require placing your finger on a fingerprint reader (such as face recognition systems) and that some biometric samples may be obtained without any action on part of the subject is further food for thought because one's biometrics may be acquired without knowledge or authorization.

Communications Security Between Sensors, Matchers, & Biometric Database(s)

Although as important as the previous issue, communications security between sensors, matchers, and biometric databases is easier to provide than to solve the problems of mutual authentication of humans and biometric sensors. Well-designed and well-implemented secure cryptographic protocols may provide the required security for sensitive data exchange between parts of a biometric identification or verification system, and they are unlikely to be the weak link in the biometrics chain.

Accuracy

A biometric system must be reasonably accurate—otherwise why would we need it? The widely used FAR and FRR, and their product, CER, are not really exact measures but often estimates made using assumptions—and these assumptions may not be reasonable in all circumstances.

Speed

Although the question of how fast the system works may not be a pressing issue in, say, a nuclear reactor access control system, it will be a crucial factor at installations such as airports or border crossing points where a large number of people needs to be reliably and quickly identified and authenticated.

Scalability

Biometric verification systems are significantly and inherently more scalable than biometric identification systems particularly because only one-to-one matching is required. A distributed, combined system using smart cards that store the owner's biometric template and compare the provided biometric in card is an example of a scalable distributed biometric verification system. However, as the previously described face recognition system experiences at airports show, system properties such as FRR must be considered in context—one false rejection a month may be acceptable, but a hundred false rejections a day clearly would not. Another scalability issue is the nature of biometrics. A scalable biometric—such as the iris—can theoretically be deployed on a large scale (with thousands or millions of enrolled users), but a biometric with weak scalability could provide acceptable error rates and performance only in small installations. Therefore, scalability is directly linked with the particular type of biometric used, and this seems to be accounted for by the International Civil Aviation Organization (see the section "Biometrics and Passports").

Resilience

A biometric system should be able to handle exceptions. An exception in this context might be a person without the required biometric or a person whose biometric may not be usable for some reason. In many cases exception handling means resorting to a manual process, which of course brings all the issues of human intervention (speed and social engineering, to name only two) with it and may mean life or death for a particular system or application.

Cost

Because laws of economics apply to almost every human activity, a biometric system should be reasonable in cost. Of course reasonableness of cost is a very subjective concept and would vary greatly between different environments and different uses.

Privacy

As mentioned in the beginning of this article, biometrics is argued to be one of the threats to privacy and anonymity in the modern age. The *Electronic Frontier Foundation* (EFF) lists the following as being the most important privacy concerns:

- Biometric technology is inherently focused on individuals and interfaces easily to database technology, making privacy violations easier and more damaging.
- Biometric systems are useless without a well-considered threat model.
- Biometrics are no substitute for quality data about potential risks.
- Biometric identification is only as good as the initial ID.
- Biometric identification is often overkill for the task at hand.
- Some biometric technologies are discriminatory.
- Biometric systems accuracy is impossible to assess before deployment.
- The cost of failure is high.

Indeed it is very depressing to imagine a society—or even worse, a world order—where everyone is forced into a biometric database and total control over all your actions and whereabouts during your entire life is maintained—and where you can never "change your username" or "log out." One cannot help but remember Benjamin Franklin's immortal statement that those who are willing to trade liberty for security deserve neither. However depressing, this image hopefully will not materialize—and to achieve that, biometric systems should provide reasonable privacy and specific use guarantees to the enrolled subjects; in addition, they must have effective systems of checks and balances to audit and assure conformance with these guarantees.

Standards in Biometrics

As Andrew Tanenbaum once supposedly said, the good thing about standards is that there are so many to choose from—regardless of whether he did or not, this statement perhaps does not yet seem to apply to biometrics standards.

- The *Common Biometric Exchange File Format* (CBEFF) describes a set of data elements necessary to support biometric technologies in a unified way, and provides for the exchange of security, processing, and biometric data in a single file. The U.S. *National Institute for Standards and Technology* (NIST) describes CBEFF as facilitating interoperability between different systems or system components, forward compatibility for technology improvement, and software/hardware integration[26].
- *BioAPI and Human Authentication API.* BioAPI and HA-API efforts merged in 1999 under the umbrella of the BioAPI Consortium. The current version of the BioAPI Specification is Version 1.1, which aims to provide a "standardized *Application Programming Interface* (API) that will be compatible with a wide range of biometric applications and a broad spectrum of biometrics technologies"[27].
- The Open Group's *Human Recognition Services* (HRS) is a module of the *Common Data Security Architecture* (CDSA), which in particular is used in Apple's Mac OS X. HRS is compatible with the CBEFF and, thanks to the CDSA modular and layered approach, can use services provided by other CDSA modules[28].
- *Biometrics Management and Security for the Financial Services Industry* (ANSI X9.84-2000) specifies minimum security requirements for effective use of biometrics data in the U.S. financial services industry, including collection, distribution, and processing of biometrics data. In particular, it specifies the security of the physical hardware used throughout the biometric life cycle; the management of the biometric data across its life cycle; the use of biometric technology for verification or identification of bank clients and employees; and other aspects. The data objects specified in X9.84 are compatible with CBEFF[29].
- The *American Association of Motor Vehicle Administrations* (AAMVA) *Driver's License and Identification* (DL/ID) standard provides a uniform way to identify holders of driver license cards within the United States and Canada. This standard specifies identification information on drivers' license and ID card applications, provides for inclusion of fingerprint data, and is compatible with BioAPI and CBEFF[30].
- *ANSI/NIST Data Format for the Interchange of Fingerprint, Facial, Scar Mark, and Tattoo Information* (ANSI/NIST-ITL 1-2000). This standard defines the content, format, and measurement units for the exchange of the specified information that may be used for identification of persons, and it is mainly directed at U.S. law enforcement agencies and government.[31]

Additionally, one of the groups of the *International Organization for Standardization* (ISO) is working toward inclusion of biometrics specifications in the widely used ISO 7816 standard for smart cards (Part 11: personal verification through biometric methods)[32].

Practical Uses of Biometrics

Because there may be as many practical uses of biometrics as users, we address just two of them: the use of biometrics for network authentication and the use of biometrics in international travel documents.

Biometrics for Network Authentication

As we saw earlier in this article, the accepted and widely used *what you know* and *what you have* authentication methods are not always—nor are they necessarily—secure or convenient, and they have their share of weaknesses.

The additional challenge of using biometrics for network authentication is the fact that the subject and the object of access are separated by a (usually uncontrolled, untrusted, and possibly hostile) network, which does not add to the simplicity or security of the system as a whole. As illustrated by the case of gelatin fingers described earlier, the question of whether a live person provided the biometric to a remote biometric sensor is even more important in network authentication applications when there are no preventive or detective controls, such as a watching guard, in place.

Although we have relied mostly on passwords to serve as the only or the main authentication mechanism until today, it has been clear for a while that passwords do not provide strong authentication. Keeping this lesson in mind, a biometric network authentication system should not depend solely on biometrics but should use one of the other authentication methods (what you know or what you have) as well.

The remote biometric sensors required in any biometric network authentication system are one of the most vital parts of the entire system, yet they are most vulnerable ones as well. For our purposes, we define the remote part of a centralized network authentication system as including a human user who needs to be authenticated as being physically present at the site and time of authentication, a general-purpose computer running a general-purpose operating system, and a special-purpose biometric sensor device directly connected to the general-purpose computer. This setup, therefore, includes the following high-level potential points of attack:

1. User
2. Path from the user to the sensor
3. Biometric sensor
4. Path from sensor to the general-purpose computer
5. Network
6. The central database

Even if the central authentication database is left out of the picture, the most simple risk assessment would reveal, among others, the following issues:

1. The user should be accurately identified or the declared identity should be verified; the sensor should be able to differentiate between a live human being providing live biometric and a biometric replica, such as an iris photograph or a gelatin finger. This includes, inter alia, reasonable assurance of the physical presence of the whole individual and not just the particular biometric at a particular point in time (hence, in part, the need for multimodal authentication involving not only what you are but also what you know or what you have).
2. The sensor should be sufficiently tamper-proof to withstand a defined set of attacks by a defined class of attackers, which would of course differ from environment to environment.
3. The communication protocol used between the sensor and the general-purpose computer should be simple, well-defined, and verified.
4. The role of the (untrusted) general-purpose computer and its software in such a system should be kept to a minimum. The biometric data acquired by the sensor should be cryptographically protected (encrypted and signed with the device key, for instance) inside the same sensor, without any dependence on action or inaction of the general-purpose computer. Their only role in this play should be to relay the bits from the sensor to the central authentication server for verification. Confidentiality and integrity of the biometric data should not be affected by a malicious, general-purpose computer or its software; the worst that can happen is the nondelivery of such data to the central authentication database.

An example of this approach would be a tamper-resistant fingerprint reader able to accurately recognize live human fingers (and reject fake ones), extract the required information, append a time stamp from an internal independent time source, encrypt and sign the resulting minutiae + time stamp data block using some digital signature algorithm, and send the resulting information through, for example, a *Universal Serial Bus* (USB) connection to the general-purpose computer. The general-purpose computer may then use the provided token to seek authentication from the central authentication database, provided all other requirements have been met.

Today a variety of network authentication systems that use or can use biometrics are available from numerous vendors. Aside from the objectively subjective information provided by vendors of such systems, little evidence of assurance exists that could enable potential users to evaluate them for their particular environments. The fact that most of these systems run as applications on the most widespread and arguably the least secure of operating systems perhaps speaks for itself.

Biometrics and Passports

For many years now more than 110 nations have issued machine-readable travel documents (mainly passports and visas) that conform to the *International Civil Aviation Organization* (ICAO) standard 9303. ICAO, a United Nations specialized agency, in addition to being responsible for international civil aviation matters, is also mandated to develop and adopt international standards on customs and immigration documents and procedures under the Chicago Convention. These machine-readable travel documents include a two-line area printed in *Optical Character Recognition* (OCR) B format, which contains information usually required for international travel (such as a person's name, date of birth, citizenship, document validity dates, and other information). These documents have greatly reduced the time necessary to check passports and visas by border officials, and have contributed to smoother international travel. In May 2003, the ICAO adopted a set of documents on integration of biometrics into machine-readable passports, choosing three most suitable for these purposes[25]. The main biometric chosen was a digitized face image, followed by two optional biometrics: fingerprints and irises. The ICAO also selected high-capacity, contactless smart cards as the storage method for this biometric data and gave other recommendations related to integration and use of biometrics in passports and other documents. It remains to be seen if or how and when 188 member states of the ICAO will integrate biometrics into their passports.

New Biometrics

It would be unreasonable to assume that we are aware of all possible biometrics. It may very well be the case that new biometrics are discovered and possibly, in the fullness of time, considered fit for practical use. An example would be a behavioral biometric proposed by Ross Anderson of Cambridge University, author of the already classic *Security Engineering:*

> "Are there any completely new biometrics that might be useful in some circumstances? One I thought up while writing this chapter, in a conversation with William Clocksin and Alan Blackwell, was instrumenting a car so as to identify a driver by the way in which he or she operated the gears and the clutch."

Summary

Biometrics is a promising and exciting area, where different disciplines meet and provide an opportunity for a more secure and responsible world. However, the same biometrics, if misused or poorly engineered, may instead bring many hassles—if not troubles. Some biometrics are less usable than others, and different environments warrant different biometrics and design considerations. The best advice would be to differentiate between market-ready biometric technologies and technologies that are not yet (if ever) ready for deployment outside testing grounds. However much fervent proponents and keen vendors of biometric solutions market their wares, the guiding factor should be proven reliability and appropriateness of these solutions to specific uses, not marketing hype, which seems at times to dominate this arena.

Organizations and Publications

The following organizations and publications may be useful sources of further information on biometrics and biometric applications:

The International Biometric Society: **www.tibs.org**

Biometric Consortium: **www.biometrics.org**

BioAPI Consortium: **www.bioapi.org**

International Biometrics Industry Association: **www.ibia.org**

International Association for Identification: **www.theiai.org**

Journal of the International Biometric Society: **stat.tamu.edu/Biometrics**

Biometric Digest: **www.biodigest.com**

Biometric Technology Today: **www.biometrics-today.com**

Additionally, the following books may serve as good introductions to biometrics:

Guide to Biometrics, by Bolle, Connell, Pankanti, Ratha, Senior, ISBN 0-387-40089-3, Springer Verlag, 2003

Practical Biometrics, Julian Ashbourn, Springer Verlag, 2003

One of the best publicly available works on security engineering is *Security Engineering: A Guide to Building Dependable Distributed Systems,* by Ross Anderson (Wiley, 2001).

References

[1] **http://www.dhs.gov/dhspublic/interapp/editorial/editorial_0333.xml**

[2] **http://news.bbc.co.uk/2/hi/americas/3358627.stm**

[3] **http://www.usatoday.com/tech/news/techpolicy/2003-08-24-biometrics-travel_x.htm**

[4] **http://www.anu.edu.au/people/Roger.Clarke/DV/HumanID.html**

[5] "On the individuality of Fingerprints. Pankanti," Prabhakar, Jain; *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* December 2001.

[6] "The History and Development of Fingerprinting," Lee, Gaensslen; *Advances in Fingerprint Technology,* CRC Press, 1994.

[7] *Guide to Biometrics,* Bolle et al., Springer Verlag, 2003.

[8] "Fingerchip: Thermal Imaging and Finger Sweeping in a Silicon Fingerprint Sensor," Mainguet, Pegulu, Harris; *Proceedings of AutoID 99,* October 1999.

[9] "Low-power and high-performance CMOS Fingerprint Sensing and Encoding Architecture," Jung, Thewes, Scheiter, Gooser, Weber; *IEEE Journal of Solid-State Circuits,* July 1999.

[10] "Ultrasound Sensor for Fingerprint Recognition," Biez, Gurnienny, Pluta; *Proceedings of SPIE—Optoelectronic and Electronic Sensors,* June 1995.

[11] "A Tree System Approach for Fingerprint Pattern Recognition. Moayer," Fu; *IEEE Transactions on Computers,* C-25(3).

[12] *Guide to Biometrics,* Bolle et al., Springer Verlag, 2003

[13] **http://www.itu.int/itudoc/itu-t/workshop/security/present/s5p4.pdf**

[14] **http://www.schneier.com/crypto-gram-0205.html#5**

[15] "Face Recognition: Features versus Templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 12(10), October 1993.

[16] *Beyond Fear: Thinking Sensibly about Security in an Uncertain World,* Bruce Schneier; Copernicus Books, 2003.

[17] **http://www.theregister.co.uk/content/archive/25444.html**

[18] **http://www.theregister.co.uk/content/archive/26298.html**

[19] "A Hand Shape Identification System," Biometric Systems Lab, **http://bias.csr.unibo.it/research/biolab/hand.html**

[20] U.S. Patent 3,576,537; U.S. Patent 3,648,240

[21] "Iris Recognition: An Emerging Biometric Technology," Wildes; *Proceedings of the IEEE,* 85(9), September 1997.

[22] **http://www.schiphol.nl/schiphol/privium/privium_home.jsp**

[23] "Automatic On-line Signature Verification," Nalwa; *Proceedings of the IEEE,* 85(2), February 1997.

[24] "Speaker Recognition," Campbell, in *Biometrics: Personal Identification in Networked Society,* by Jain, Bolle, Pankanti, ISBN 0-7923-8345-1, Kluwer Academic Publishers, 1999.

[25] **http://www.icao.int/mrtd/download/technical.cfm**

[26] **http://www.itl.nist.gov/div895/isis/bc/cbeff/CBEFF010301web.PDF**

[27] **http://www.bioapi.org/**

[28] **http://www.opengroup.org/security/cdsa.htm**

[29] **http://www.ansi.org**

[30] **http://www.aamva.org**

[31] **ftp://sequoyah.nist.gov/pub/nist_internal_reports/sp500-245-a16.pdf**

[32] **http://www.iso.org**

[33] **http://news.com.com/2100-7348_3-5158973.html**

The author of this article does not work for, is not affiliated with, and has no financial interest or shareholding in any vendor of any biometric technology at the time of submission of this article for publication.

EDGAR DANIELYAN, CISSP, is a self-employed consultant, published author, editor, and instructor specializing in information security, UNIX, and internetworking. He is the principal partner at Danielyan Consulting LLP (**www.danielyan.com**), an information security assurance consultancy, and a member of ACM, IEEE, ISACA, USENIX, and the British Computer Society's Information Security Specialist Group.
E-mail: **edd@danielyan.com**

Book Reviews

The Unicode Standard

The Unicode Standard, Version 4.0, by The Unicode Consortium, ISBN: 0-321-18578-1, Addison Wesley Professional, 2003.

The Unicode 4.0 book is a thick, heavy one, but it is good. If you work with the Unicode character set, you should have this book on your bookshelf.

This book consists of four parts:

- Background and explanation of terms (103 pages)
- Implementation guidelines (29 pages)
- Technical specifications (60 pages)
- The Unicode Character Tables (1150 pages)

A review must describe each of these sections by itself, because they are important for different reasons. Unfortunately, however, the sections in the book are not clearly divided into sections as I outlined, so you don't necessarily know where to start. You don't need to read the characters section—just the sections you are interested in.

You should read the "Preface" (Section 0), because this section describes the rest of the book. It starts on page xxxi (before chapter 1).

You can then immediately go to the section you are interested in. Each section more or less stands by itself, and the book is easy to read. If something is not clear, you should look for text in another section that describes the subject. Reading from start to finish is possible, but I use this book as reference material, like an encyclopedia (except for the characters).

The background material is easy to read. It covers basic concepts such as differences between *characters* and *glyphs*, definition of terms such as *equivalence*, character encoding schemes and implication of things such as bidirectional text (mixed right-to-left and left-to-right text). Knowing how these things work is essential for anyone who either implements text engines of any kind or works on developing protocols or standards. This background material is easier understood read on paper and not electronically. It also is the part of the book I return to most often.

The second very good part concerns implementation guidelines. Even though it is (relatively) short, it is very important material. It discusses selection algorithms and other user interface guidelines, as well as other algorithms needed for, for example, comparison (what is called "Normalization"). I like this section as well, because it really describes the details you need to know when implementing anything Unicode related.

Unicode is a *large* character set. You see that in the more-than-1000 pages of "just characters." Of course, the tables themselves can be found on the Unicode Consortium Web site, but this book gives you a good overview. Part of this overview is a description of the *scripts* that Unicode covers, one at a time before the *codepoints* that come from those scripts. Still, this is the part that makes this book heavy, and a version without the codepoints would have been interesting by itself.

The book ends with more technical material, consisting mostly of references to, for example, *Unicode Technical Notes* and other standards documents that the Unicode Consortium produces, in addition to the Unicode Standard itself.

Useful reference

In summary, the first 130 pages (well, starting at page 40) in the book are very good. If you work at all with Unicode, you should read those pages. The rest of the book is good reference material.

Even though I have been working with Unicode for almost 10 years now, and for the last 8 years have weekly reviewed Unicode-related standards in the Internet Engineering Task Force, I see myself opening this book now and then. There is always something I need to check, and to be honest, I like encyclopedias on paper.

As reference material, this is a must-have item. If you want to read only the 140 interesting pages once, well, the book is possibly overkill.

—Patrik Fältström, Cisco Systems
paf@cisco.com

---

iSCSI: The Universal Storage Connection

iSCSI: The Universal Storage Connection, by John L. Hufferd, ISBN 0-201-78149-X, Addison-Wesley, 2003.

I have to come clean straightaway and say that when I received this book to review I had never even heard of *Internet Small Computer System Interface* (iSCSI) and, to be honest, I have never heard it mentioned by anyone again since the day the book arrived. This is, of course, not a criticism of this book, just a comment on the current state of penetration of iSCSI into everyday computing discourse. In fact, if you search Google for "iscsi," you get only 465,000 hits—very few indeed these days, though this does have the decided advantage that the links you get are generally pretty useful. I'm sure that this will change because there are lots of big names behind the protocol, and certainly when vendors start really selling kit that uses it. *Storage Area Networks* (SANs) are important (though also not yet at the forefront of most computing people's minds)—and iSCSI will probably make them bigger.

However, to the book. And, really, if you want to know pretty well everything about iSCSI and don't want to read lots of Web sites, then this book is for you. It covers everything from the background behind the protocol, to how and where it might be applied, to all the low-level information that most of us hope that we never need to see. I'm not going to list it all and go into detail: the whole thing is here, from soup to nuts.

As to the presentation of the material, it is excellent—clear diagrams and useful tables. The layout is spacious without huge amounts of wasted white space on every page—making a change from many textbooks you see today.

The writing is clear too, though I did find myself becoming a bit bogged down in all the abbreviations (no, not acronyms—most of them are not words), which seem to pile up in the sentences. I got a bit tired of seeing iSCSI everywhere after a while too. I wasn't keen on the end-of-chapter summaries, finding them a bit redundant.

Good Reference

All in all, if you are in a position where you need to know about iSCSI and may have to be involved in working with it at a low level, this book is a good reference. I doubt that there is anything more comprehensive or better written at the present time.

—Lindsay Marshall, University of Newcastle upon Tyne
lindsay.marshall@newcastle.ac.uk

Read Any Good Books Lately?

Then why not share your thoughts with the readers of IPJ? We accept reviews of new titles, as well as some of the "networking classics." In some cases we may be able to get a publisher to send you a book for review if you don't have access to it. Contact us at **ipj@cisco.com** for more information.

Fragments

NRO Comments Concerning ICANN and WSIS

The *Number Resource Organization* (NRO) is the coalition of *Regional Internet Registries* (RIRs) which operate in the world today. The NRO is an organization representing the collective experience of individual RIRs and their communities. While the prime subject of its work are matters of joint interest relating to Internet numbering resources, the NRO provides an efficient interface to other parties interested in these issues. As the Internet continues to evolve, the NRO will ensure continuity of the operational infrastructure of Internet number resource allocation.

The RIRs are responsible for distribution of *Internet Number Resources* [IPv4 and IPv6 addresses and *Autonomous System Numbers*]. These number resources are the most fundamental of the identifiers on which the Internet relies: the Internet can operate without domain names; but it cannot operate without numbers. The RIRs have carried the responsibilities associated with managing these critical resources collectively for over 10 years, since well before the start of ICANN. This has been done very effectively through the entire "modern history" of today's Internet which includes both the "dot com boom" and the "dot com bust."

The RIRs have participated in the *World Summit on the Information Society* (WSIS) processes for over a year, including regional Prepcoms and the Summit itself. This is probably longer than any other Internet organization. The RIRs have attended as observers, and as subject matter experts with a genuine aim to assist in debates and discussions around issues related to Internet Number Resources in general and to IP addresses in particular.

The RIRs participated in the WSIS Phase I process as full supporters of ICANN as the model which represents not only the fundamental and critical aspects of Internet development to date, but also the means of community self-regulation to administer and manage Internet Number Resources. It must be understood that this is not given by the RIRs as mere components of ICANN, dependent upon it for support; but rather as independent components of the broader Internet administrative framework which ICANN itself is intended to support.

In the second round of WSIS, the NRO speaking for the collective RIRs will assert an active role vis-à-vis ICANN in order to aid that organization to address the genuine questions that it faces. The principle of these issues within the WSIS context is that of the independence and genuine internationalization of ICANN.

Therefore the NRO calls on ICANN to continue its work in this area, not by building a multinational organization, but rather by including and gaining the genuine support of its significant base of core stakeholders, namely those in the DNS, IP address, and protocol communities. Furthermore, the NRO calls on ICANN to work with the US Government to demonstrate a genuine and unambiguous plan for its independence and to commit to this plan before the conclusion of the second phase of the WSIS.

Finally, the NRO rejects any concept of an alternative Internet administrative model located within any governmental or intergovernmental structure. The NRO acknowledges that there is a valid role for governments in the administration of the Internet but this must be in the context of the current model. There is a need for the continual improvement of the current model of industry self-regulation to the extent that the ultimate solution may look little like today's ICANN.

http://www.apnic.net/index.html
http://www.arin.net/index.html
http://www.lacnic.net/
http://www.ripe.net/index.html

Upcoming Events

INET/IGC 2004 will be held in Barcelona, Spain, May 10–14, 2004. INET, which is the annual conference of the *Internet Society* (ISOC), will this time be held jointly with Spain's *Internet Global Congress* (IGC). For more information, visit: **http://www.isoc.org/inet04/**

The *North American Network Operators' Group* (NANOG) will meet in San Francisco, May 23–25, 2004. For more information see: **http://nanog.org/**

The *South Asian Network Operators Group* (SANOG) will meet 23–30 July, 2004 in Kathmandu, Nepal. More info at: **http://www.sanog.org/**

The *Internet Corporation for Assigned Names and Numbers* (ICANN) will meet in Kuala Lumpur, Malaysia, July 19–23, 2004, and in Cape Town, South Africa, December 1–5, 2004. For more information see: **http://www.icann.org**

The *Internet Engineering Task Force* (IETF) will meet in San Diego, CA, August 1–6, 2004 and in Washington, DC, November 7–12, 2004. For more information, visit: **http://ietf.org**

The *Asia Pacific Regional Internet Conference on Operational Technologies* (APRICOT) will be held Feburary 16–25, 2005 in Kyoto, Japan and February 15–24, 2006 in Bangalore, India. For more information visit: **http://www.apricot.net/**

The Internet Protocol Journal

Ole J. Jacobsen, Editor and Publisher

The Internet Protocol Journal is published quarterly by the Chief Technology Office, Cisco Systems, Inc.
www.cisco.com
Tel: +1 408 526-4000
E-mail: ipj@cisco.com

The Internet Protocol Journal, Cisco Systems
170 West Tasman Drive, M/S SJ-7/3
San Jose, CA 95134-1706
USA

ADDRESS SERVICE REQUESTED

PRSRT STD
U.S. Postage
PAID
Cisco Systems, Inc.