It's the Anthropology, Stupid!

Ross Anderson and Frank Stajano

Imagine a world five or ten years from now where virtualisation has become pervasive. Rather than doing your work on a personal computer, you have a laptop (or a tablet or a virtual reality headset) with a number of virtual machines – say one for work, one for play, one for serious personal things like banking, and one for the classified work on the defence contract your employer picked up.

These virtual machines communicate in turn with a number of virtual servers in various clouds. Your work machine talks to a corporate cloud and to services run by subcontractors such as safesforce.com. Your play machine might, according to taste, talk to an assortment of game servers, porn servers or bridge clubs. Your serious personal machine will talk to your bank; you might have other machines to talk to other banks, or to your broker, or to your oncologist, but for the sake of argument let's just consider one of them for now. And your classified machine will talk to a dark green box somewhere in the government cloud. What's more, many of these servers will not be single VMs but will be replicated over several machines – maybe in more than one country for resilience.

This brave new world is the direction in which both hardware design and service delivery appear to be travelling. What extra work will it create for the security engineer? Well, there are some matters of functionality and assurance at the operating system level. First, we need better separation between virtual machines, so that different VMs on the same physical machine can't attack each other. Second, high-assurance service providers will want ways to tie down particular services to particular sets of boxes; and third, they will want robust ways to get key material into VMs. Whether these two problems are solved using administration or some kind of hardware cryptoprocessor is really a detail. Surely the virtualisation vendors can harden up their product, and surely service firms will work out ways to manage keys. And whether communications with client VMs are handled using TLS, or SSH, or Kerberos, or something new, is really not that exciting. Of course there will be plenty bugs and blunders, but basic protocols are no longer rocket science.

When we contemplate the client end, things get harder. The modern laptop or desktop PC is a much more complicated beast than a typical server, and providing high-grade technical separation between VMs is not at all trivial. For example, how do you assure yourself that only you have access to the microphone and camera right now? And what about covert channels in the graphics card? There is some red meat here. Presumably the services with most to lose by being excluded from using commodity platforms – yes, the military – will fund the necessary research, as they have in the past.

But perhaps the biggest problem of all will not be the cryptographic protocols that tie one virtual machine to another, but the human protocols that enable the right user to deal with the right virtual machine.

To Whom am I Speaking?

Once our typical user has multiple VMs in his laptop – alice.work, alice.play, alice.bank and alice.mil – the traditional view is that she needs a trusted selection mechanism to help her invoke the right VM for the job. The enemy intelligence agent, Charlie, will try to trick her into mistaking a mil-like simulation in his game site in alice.play for the genuine alice.mil. So we will engineer a secure attention sequence, control-alt-delete or whatever, to take Alice dependably to a hypervisor where she can select from among these four machines. We will then train her to never enter classified information without going through this ritual first.

So far so good. But what about her brother Bob, who has no clearance? As a result, he's untrained in the holy doctrine of trusted path and has not been indoctrinated into the ceremony (indeed the reflex) of the secure attention sequence. There's no bob.mil; in his case the fraudster David is trying to trick him into mistaking a bank simulation in bob.play for bob.bank.

And what about mistakes? They matter much more than targeted attacks. To a first approximation all attacks are by insiders, and most of them spring (initially at least) from errors rather than malice. The main practical benefit of mandatory access control is that it prevents Alice entering High information into a Low system by accident. Accidents also cause much expense and embarrassment to civilian users. (One of the first writer's most embarrassing moments in recent years came while using a PC at a friend's house to check email, and – on an unfamiliar browser – deleting the host's cookies, passwords and browsing history, while trying to unjam gmail.) Mistakes are often caused by getting the context wrong, so if we're going to make them less likely, our designs should be better at synchronising the user's mental model better with that of the machine.

This brings us to the central idea of this paper. It's not just computers that operate a number of virtual machines; people do as well. The human-computer authentication problem isn't one-way, but two-way. It's not enough to tell Alice or Bob "hereafter you will be in machine.work"; a robust system must also assure the machine "the user sitting in front of you is in the mindset .work".

In other words, secure virtualisation isn't just about ensuring that the right VM in the laptop talks to the right VM in the cloud. It's about ensuring that the right VM in the laptop (or the cloud) talks to the right VM in the user's brain. It's not primarily about the outside attacker, but the insider: and the critical question is which insider.

Psychologists have written about how our different personae fail to anticipate each others' likely reactions. For example, Dan Ariely and George Loewenstein showed that when we are in a rational or 'cold' state we underestimate how we will react when in an aroused or 'hot' condition; with experience, we learn to avoid circumstances in which we might fall into temptation and lose our temper, drink too much, or spend more than we should [1]. But the strategies people develop to cope with this "hot-cold empathy gap" don't always translate so well online. As another example, Joe Bonneau and Sören

Preibusch remark that Facebook might make people more vulnerable to scams, because it encourages rapid browsing and instant interaction in a noisy, distracting environment while often in a state of partial arousal [2]. This has clear hazards if Facebook users operate payment mechanisms, or authenticate themselves to other websites, while in a careless mode – in other words, if alice.play talks to machine.work or machine.bank.

The traditional solution to the 'human VM' problem involves the empathic synchronisation we achieve via nonverbal communication, to which cultures add manners, rituals and dress codes. Soldiers wear uniforms, and the parade-ground ritual of calling them to 'attention!' literally gets their attention for the orders of the day. (It's the military equivalent of control-alt-delete.) The banker's suit, and the dentist's white coat, condition the behaviour of the wearer as well as his customers or patients.

How then can we bridge the gap between the mechanisms that we can use to order human VMs, and the quite different mechanisms that work with the technical variety?

A Modest Proposal

Matt Blaze noted how many real-world ceremonies, such as the ritual of ordering and sampling wine in a restaurant, are actually security protocols [3]; and Carl Ellison argued that human-computer protocols should be seen as ceremonies, as the act of typing a password or shoving a smartcard into a reader are interpreted as such by users [4]. This seems sound; we need to recapture the anthropological aspect. Another useful idea was the comments here in 2004 by Frank Stajano that users wear different hats when operating a PDA in different modes [5].

Our mission is not to tell the user which VM in the laptop she's talking to, though. It's to tell the laptop which user VM it's talking to. We have to turn things round.

The simplest proposal is that the user should wear a physical hat signifying her mode of operation. In order to access alice.mil, Alice should put on her service cap, which will be recognised by the laptop's camera; to log on to his bank, Bob will put on a banker's bowler hat. For a .play machine, a party hat is fine.

There are other ways in which this could be implemented. For example, to access your work machine, you might just put your work ID badge round your neck. And of course, there will be a temptation to shift everything to software – so that the hat you're wearing is simply reflected back to you in an icon on the screen. But what I'm arguing for here is a change in perception. The objective is to be sure that the user is in the right frame of mind, not that the user 'can' find out what VM she's talking to. (And as for the worry about whether you have the wrong user, I'm not interested in that here.) So rather than thinking about the problem as a system engineering one, I'd rather think of it as a somewhere between applied psychology, biometrics, and anthropology.

Interesting possibilities arise from the work of Peter Robinson and others in affect-based computing [6]. Now that computers are routinely equipped with video cameras and microphones, a machine can study its operator's facial expressions, gestures, speech and posture to observe emotions such as impatience, perplexity, and anger. This new field also has some useful warnings for the engineer; for example, people's emotional reactions to robots are very different from their reactions to people, so it cannot simply be assumed that having a hat icon on-screen will have the same effect on Alice as wearing a hat on her head. The social aspects matter; so do the physical aspects. The old world of rituals and manners, which condition our behaviour and our expectations of others, is different from online. If homo interneticus spends twelve hours a day sitting on a sofa with a laptop, much that used to go into human interaction will be lost. In fact, we don't even have a proper word for this loss!

In conclusion, I argue that to build robust protocols that are good enough to authenticate virtual machines to each other, we must first understand that the VMs in question include those in the brain as well as in the software, and that the authentication must be mutual. Protocol design thus comes down to usability testing, and something a bit more. Maths and formal methods give way to experimental psychology. But ultimately, at the deepest level, this is about anthropology. It's how systems get embedded in culture, in the interplay of verbal and non-verbal communications, and how our evolutionary past conditions the way we behave. The physical, and the ritual, have value that we need to rediscover. Or perhaps the next generation will reinvent it anyway, in the same way that pidgins become creoles, and all we have to do is figure out how to provide the avatars – or even just a basic lexicon of gestures to reopen the nonverbal channel.

Further reading

[1] Dan Ariely, George Loewenstein, "The Heat of the Moment: The Effect of Sexual Arousal on Sexual Decision Making," Journal of Behavioral Decision Making v 19 (2006) pp 87–98

[2] Joe Bonneau, Sören Preibusch, "The Privacy Jungle: On the Market for Data Protection in Social Networks", Workshop on the Economics of Information Security 2009

[3] Matt Blaze, "Toward a broader view of security protocols," 12th Cambridge International Workshop on Security Protocols, April 2004

[4] Carl Ellison, "Ceremony Design and Analysis', IACR eprint 399

[5] Frank Stajano, "One user, many hats; and, sometimes, no hat – towards a secure yet usable PDA", 12th Cambridge International Workshop on Security Protocols, April 2004

[6] Peter Robinson and Rana el Kaliouby, "Computation of emotions in man and machines", Phil Trans Roy Soc B vol 364 (Apr 2009)