IMS : The Best Protection Against Identity Theft

A highly opinionated commentary by Mike Gonzales at IBM's Silicon Valley Lab

I received a letter in the mail the other day. Yeah, the snail kind. It was from my health care provider and it warned me that a computer was stolen from one of the medical group offices. The computer contained my identification along with the ID information for a bunch of other members of the health care provider. The letter went on to advise me that I should contact the credit reporting agencies and my credit providers to alert them and to take preemptive measures to prevent my identity from being stolen. What are the odds that I'd already have a pending credit card application? Okay, well, actually the odds are pretty good since I'm always looking for a better deal. So what should I do? Pity the poor slob who tries to be me? Nah. How about I laugh at all the bad decisions that resulted in my identification ending up on somebody's laptop? Yeah, maybe that's my tack.

It got me to thinking about whether or not there's been a case of identity theft when a mainframe computer is stolen. Could it be? Go ahead, steal my big blue iron. I'll be mad and out of work for a while but all you'll get is a pricey box of parts. You won't get any decipherable data with my mainframe. It's just a bunch of cool fast CPUs, more memory than I can use and an I/O cage with enough channels to support more DASD than I can fit on the raised floor.

Okay, so you know about mainframes and figured it would be easier to just pop a drive out of the DASD farm. Admittedly, the farm gets smaller as the DASD gets bigger on the inside and better (in a shrinking footprint!), but even so, you'd have to luck onto the right set of boxes if you want to get any meaningful data. Maybe you don't have a truck or you're a savvy techie and you know you can pop out just one of the many drives from a DASD box. But which one? I put my data all over the place and I'm not even trying. Those 2105s and 2107s spread my files all over the place an in record time! With RAID anything and large volumes you have pieces of data spread all over the place so you'd better be good at figuring out exactly which drives have the data you want because you have a lot to pick from. In *every* DASD box. Behind secured and alarmed doors.

And hey, what if the data on the DASD boxes is encrypted through the crypto coprocessor on the mainframe using my IMS crypto exits in ICSF? Now we're back to hauling off the mainframe so you can read the data on the DASD being hijacked. And it's going to take a sizable mainframe/z/OS/IMS-knowledgeable team and a bigger truck to even get started.

Well, maybe you think you can just take the DASD box and plug it into your SAN back home. Hmmmm. Did you remember to find and steal my keys data set, too?

It's not likely to be on the same DASD box as my databases. My own wife wouldn't have a clue on how to find and decipher the keys and she *knows* me. Good luck.

What about the physical security involved? Have you heard of a serious institution with their mainframes out front or in the administrative offices? Do you suppose you or I could stroll up to those machines, open them up and start pulling out parts and not have anyone notice?

A laptop, a desktop; they're good to have and have their place in the world of data but they just don't fit the bill for being data stores for confidential information. C'mon, I'm not the first one who has received one of these letters. The Lawrence Livermore lab was hit, UC Berkley was hit, my health care provider was hit. All small machines with big data. All prime candidates for data liability and theft. So what can you do to prevent your ID from being stolen? As a consumer, not much more than the typical recommendations like not using your credit card online. As an IT professional there's a lot you can do. Start planning on getting that data off of small machines. Plan on a central data store that's behind locked doors and accessible only through a protected storage network. Use IMS. You can't put IMS databases on small machines. Well, okay, maybe a P390 or similar ilk but would you really put production data there? I've seen lots of test data on the small machines and if you think about the risks to data that's all you'll put on the small machines, too.

The features and convenience of a lot of software can tempt you to use your laptop as a home for databases but do you think space exploration was controlled from a desktop server? No my friends, it began on a mainframe with mainframe software like Information Management System (IMS). Say the name. It's concise. A system that manages information. And it only works on a mainframe. Period. You want to use e-commerce? Go ahead! Let all the pretty front-ends do the pretty work but don't let your data out of the vault. You want distributed? Go ahead! You can have more than one mainframe in different locations. They can share data or you can duplicate it but don't leave it on a vulnerable machine. Keep it in a big box, preferably blue.

Seriously, as data and data integrity professionals we owe it to our customers and our employers to make recommendations that make sense. If the data is sensitive it shouldn't be stored on a portable computing device. It belongs on a real server that is protected physically, intellectually, and logically. Even if you don't want all the protection and power offered by a mainframe, you can still centralize your data on dedicated DASD boxes. When have you ever seen somebody walking out of the office with an armoire sized box under their arm without getting questioned?

So you have the comfort and security of your data on a mainframe. You're behind thick walls, locked doors, nobody is going to drive away with your data.

But you have to allow access to it. You have to be able to create, manipulate, and destroy data. You have to be able to manage the data. Not everybody needs it and some of it is for very few sets of eyes. You need a scheme to allow different people and groups to have varying levels of access to the data and the programs used to manipulate the data. So the next consideration has to be data access. It's great to make the data safe and secure but it's not very useful if nobody can use the data. Data that's inaccessible is just spinning bits. (That brings up questions that I haven't had answered yet; Which is heavier, a disk full of 1's or a disk full of 0's? And will the disks wobble if you put all the data on one half of each platter?)

Data that can be used is information. To use the data as information and to manage how it is used and accessed you'll need some sort of information management system. IBM has one that is appropriately named; the Information Management System.

IMS is the premier high performance secure database and transaction manager we've all come to know and love. With IMS managing your data every second you can process tens of thousands of transactions or just a few hundred, but always in a secure, recoverable, and auditable manner. How does that help with prevention against identity theft? It's the secure and auditable stuff. IMS has *lots* of security options, most through the Security Authorization Facility (SAF) interface, but don't forget the primary and long standing data protection IMS has offered since the 1960's; the protected view of data through the combined Program Specification Block (PSB) and Program Control Block (PCB).
Your database can have a lot of data needed by a program or many programs but not all programs need all the data or all the levels of data in a database. The PSB and PCB will allow a program to access only as much or as little data as you deem necessary for that program to provide service.

Add Resource Access Control Facility (RACF) or equivalent SAF provider profiles and you can build a fortress around your system. You can define protection at the network access level, subsystem level, by transaction, by database, by time of day, day of week, or any combination of the above and more! Even within an IMS program you can extend access to database fields beyond the protection provided by the PSB/PCB combination.

With RACF you can create any number of protection schemes to allow as much or as little access by as few or as many people and programs as you need. You can have some of your data behind a dozen password protected layers or one layer or none.

Phew, now the hatches are battened, keys are turned, but what impact to performance at the cost of security? Could be big, could be small. The answer depends on your data access patterns and how fat your transactions run. The impact also depends on how tightly you've secured your data and transaction

access. More layers can cost you more overhead but nothing *this* good is cheap. Fortunately there are some steps you can take to minimize some of the impact. For almost ever, RACF has given us the option to use Virtual Lookaside Facility (VLF) for caching and quickly recalling a variety of security blocks. It's not much good the first time you have to create an ACEE but it sure is nice the next time you need it!

You can also split up your RACF database into a few or a bunch of data sets to allow parallel access to the RACF DB so more requests can be processed in a shorter amount of time. And to boost that performance even more you can carve out a CF structure for each of the RACF DB data sets. It works along the same line as the VLF cache; the first time you need to read the RACF DB for a user you go to the DB and suffer the I/O, but *next* time (and there will be a next time) you get to read at CF speed. Nice!

And IMS helps with some of the ACEE caching, too. If you're an OTMA user IMS will keep your ACEE hidden away locally so we can recall it at CPU speed.

And don't forget the auditable part. You can keep a lot of tracks in the IMS Online Log Data sets (OLDS) but we all know that can slow you down a little, after all, it's a lot of I/O. Well, that can be minimized, too. *Lots* of OLDS buffers (24K OLDS blocksize works best to get those buffers above 2G) and fast DASD (like the DS8000 2107-921 currently benchmarked to log at up to 82 MB/sec) will help cut down the OLDS logging overhead.

IMS, RACF, z/OS, DS8000, System z9; bundle all these together and you can rest easy that your identity and mine will be inaccessible to all but those intended to have access.