in the cloud.

a talk about buzz words...

the REN-ISAC

- ren-isac.net/about/index.html
- historical preso's (there are many)
  goo.gl/chH2B

within our membership

- 325+ Institutions (500+ ‘distinct’ campuses, state-systems, etc)
- 825+ individual members (role is firefighting with enterprise responsibility)
- Mostly North America (few scattered throughout other english speaking countries)
- lots of ipv4 allocations
- lots and lots of ipv6 allocations (in production for years)
- big bandwidth
  - typically a few hundred meg to multi-gig pipes
  - internet2 backbone -- ~200 universities, 40-100 gig
- lots of different cultures, perceptions, ideals
- lots of diverse students (laptops coming and going from .kr, .cn, .us, .eu, .etc)
- firewalls... ha. yea right.
- Everyone and every institution is their own unique snowflake

how do you even go about sharing information?

what we tried...

- webpage with wget scripting of plain text to firewalls, ids’s and dns servers (2008)
- RT+IR+Prelude with wget scripting of plain text to firewalls, ids’s and dns servers (2009)
- people had to do their own conversion to the $DEVICE rules

what else we tried...

- RT+IR+Prelude+CIF+XML with a “special client-side library” to $DEVICE (2011)
- RT+IR+Prelude+CIF+JSON with a “special client-side library” to $DEVICE (2012)
- our library did the conversion to your $DEVICE for you (magic)

lessons learned

- plain text was too hard to parse and build into applications (no API)
- XML was too fat to parse quickly (scale--)
- JSON was too loose to to enforce any kind of “morals” on the data, serialization still took it’s toll...

protocol buffers

- Protocol buffers are Google's language-neutral, platform-neutral, extensible mechanism for serializing structured data – think XML, but smaller, faster, and simpler.
- developers.google.com/protocol-buffers

protocol buffers

- enforce the format
- portable across programming languages
- 10x-100x smaller than XML/JSON
- faster encode/decode (with scale, every second counts)
- easy integration with anything that "carries a message" (zmq, http, smtp, etc...)

now for the philosophical stuff...

what we’re working to solve over the next ten years

it is now the past...

- large mailing lists of people you’ve met at the bar and are willing [/mandated to?] share data with
- web-portals you can share data via a wiki
- web-portals you can download a pdf from
- web-portals you can download structured data from (with/with-out an actual API)

it is now the past...

- trust is controlled by how much the group is willing to share with itself
- the larger the group, the lower the overall trust measure
- there are hard ceilings to data-sharing in this model
- these are all problems we have today

what social networks have re-taught us.

- build your platform so data-hubs can grow organically within your "social graph" (or org)
- allow those hubs to be self-selective to whom they will share what types of data to (not everyone is created equal)

social network #fail

- they do not allow their hubs to interoperate with other networks (and therefor other hubs...)
- AOL made IM easy, Jabber re-invented it and took it over (till everyone moved to fb)
- we saw this movie play out over the last decade+... Prodigy is no longer with us.. :(

the next ten years

- yes, this is a ten year problem
- AOL didn't "realize" they were a "media company" till the early to mid 2000's
- it took that long for the browser market to solve this "federation" problem and gain adoption.
- it took that long for web2 to take hold

the next ten years

- teh Facebook is a social platform for connecting you with your friends
- the LinkedIn is a social platform for connecting you with your friends who have $$ and would be dumb enough to hire you
- the google plus is a social platform for security peeps who have no desire for Facebook’s shenanigans (hi ferg!)

the next ten years

- the APWG is a social platform for connecting e-crime researchers
- the US-CERT is a social platform for connecting .gov with each-other and private industry
- the REN-ISAC is a social platform for connecting edu’s with other edu’s

the next ten years

- what happens if there was a google+ feature that allowed you to specifically share something with a target group?
- what happens if you could dump structured +encrypted data in that sharing window (ever played with scrambls?)?
- what happens if there was an API into that platform...?

with technology like this

- why does the REN-ISAC need to exist at all?

where we fail

- most [international] information sharing communities are great aggregators of internally shared information
- most cross-hub action happens by those who are in many communities
- it seems like we’re actually just inhibiting the data-sharing process

you should be thinking...

- if you're not already doing automated data-sharing, why not?
- should you be focused on designing a new standard? or evolving something that already works?
- what does your architecture look like in ten years if you're standardizing around XML or JSON (or even Protocol Buffers)?

you should be thinking...

- yourself as a platform for trusted relationship building (reads: do you have a bar night at your cons?)
- how to enable your community to individually share data with the rest of the world, not just with itself
- is your business model focused on sharing data? or facilitating relationships..?

the new Science of Networks

- people are hubs
- the should be enabled as such

solve problems, don’t invent them

- what tools exist to help solve this problem?
- are your partners thinking in terms of big data?
- are you thinking in terms of big data?

collectiveintel.net

free.

(as in beer)

Skynet DEVELOPMENT TEAM