

Naming, DNS, & Security

Applied Networks and Security (IT 263 901)

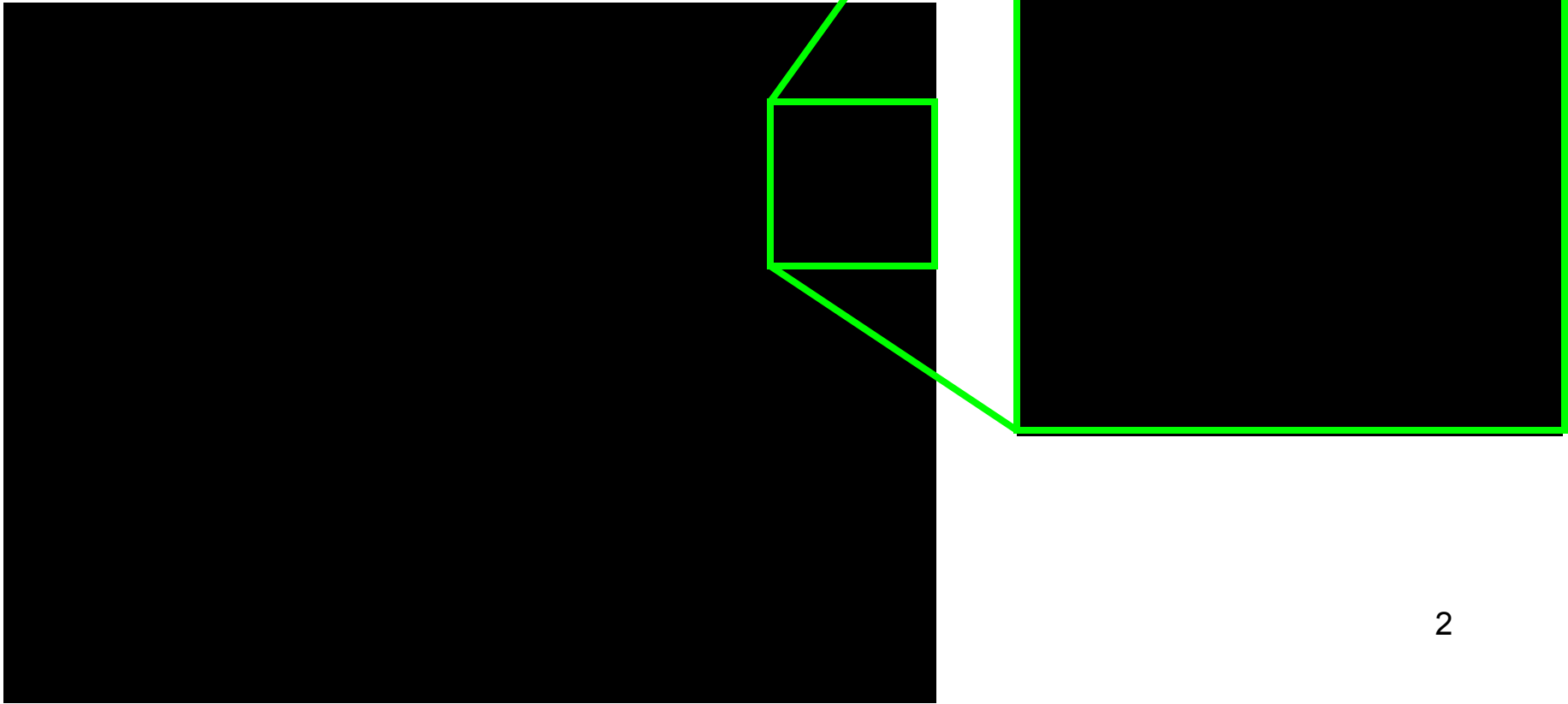
April 16, 2007

Lewis 1513

Edward Lewis

Applied Security Indeed

DePaul's application of security to the water-flow network



Seeing Chicago



Last time I taught

- Course numbers weren't 6 digits long
- The building wasn't named after me
- I used chalk to draw on the board

Hagerstown Packets

- Anyone know what the "Hagerstown Packets" are/were?
- Hagerstown's claim to fame: the place where Willie Mays played his first pro baseball.

Hagerstown

- Known as the "Hub" City
- The "Packets" were a minor league baseball team in the 1950's
- Forward looking town?
 - Hardly
 - Terminology and concepts from railroad

Evolution of Technology

- Internet built on concepts from the past
- Nothing is brand new
 - Signal technology science is old
 - Communications are older
 - The concepts behind the web date back to at least the 1940's

Keeping that in Mind

- Let's look at one portion of the Internet
- The Domain Name System
 - One of the core technologies
 - What is it? Why is it?
 - Where did it come from? Where is it going?
 - And, what does this have to do with security?

What's there to discuss?

- The data in the DNS
 - and how to secure it
- How the data is delivered in DNS
 - and how to secure it

Naming

- What do you call things?
- Why?
- Tuples

Electronic Naming

- Not "Gaming"
- Many attempts
 - X.500, LDAP, X.400
 - Apollo Computers (way back in time)
 - SS7 (phone call routing)
 - DNS
- DNS is the Internet's
 - It is the winner of many attempts

DNS Naming

- Hierarchical
 - A root name is the "parent" of all
 - Names are connected in a tree
 - Meaning only one "path" from the root to any name
 - Names are really addresses, not they are routes
 - Huh?

Names, Addresses, Routes

- Name: sticks with the object
- Address: where the object is
- Route: path from one object to another

- All three are identifiers

DNS name

- condor.depaul.edu.
- "." - run by a mysterious arrangement of people (more later)
- edu - delegated to Educause for schools
- depaul - to what you pay tuition
- condor - a host

The name

- The name of the machine is really just condor, but we usually say the "fully qualified domain name"
- The DNS identifier is also an address in the data space
- The DNS identifier is also a route through the data space to find the machine

Why did/does DNS win?

- Speed
- Scale

DNS Speed

- It's query model is simple
- It's naming is the indexing
- It's processing is limited
- By being simple minded
 - It's fast
 - It's frustratingly "stupid"

Tuples in DNS

- DNS lets you ask - "gimme the stuff at Qname, Qclass, Qtype"
- The responses are limited
 - Answer
 - Referral to another server
 - "Ask a different question"
 - Error condition

DNS Scale

- Portions of data "space" are delegated easily
 - E.g., all ".us" names go to US gov't
- The DNS is a highly tolerant protocol
 - Built on an unreliable roadbed
 - Lazy managers can still get work done

So, what's the problem?

- DNS was started by folks with as much formal education in the Internet as y'all
- They didn't know what they started
- No responsible adults were watching
- DNS was like a new volcanic island emerging from the ocean - lush and unclaimed

root

- As all names in DNS are really routes from the root, we have to know where the root is
- Folks fool us into thinking that this isn't so important, because what matters is "com." But com is just one set of servers away from the root

Whence "the" root?

- The root we know was born in a contract between the National Science Foundation and the InterNIC.
- The InterNIC was operated by NetworkSolutions

One day...

- NetworkSolutions claimed they owned the databases of the InterNIC and the domain name business was born
 - NASA policy - it's NASA's
 - DARPA policy - it's yours
 - NSF - had no policy and that started the trouble

The Early DNS

- .com, .net, .org, .edu, .int, .gov, .mil
- generic Top Level Domains (TLD)
- Most activity was in US, Canada, a few lines to Europe, Japan, and Australia
- Most operations were through government grants

A little later on DNS

- Country code delegations were added (ccTLDs)
 - From a standard list ISO 3177, a document that changes
- Original US contracts began to end, bringing a shift in attitude about the Internet
 - From "toy" to "tool"

Power Struggles

- Behind DNS is a database of entrants used to make up the DNS contents
- Databases are very valuable things
 - Lists of potential customers
 - Knowledge is money!
- There was another asset hidden too

Location, location, location!

- Imagine a McDonalds in a corn field no where near an Interstate or highway, no people around.
 - Not too busy, not too profitable, not valuable
- Imagine a McDonalds at a highway oasis or a mall
 - Valuable

The .com database

- The .com database is the mega-shopping mall of the "Information Super Highway"
 - If you want to be in business you better be there, to be there you have to pay (now)
 - You really want to be the one collecting all that money
- Same is true for .net, .org (the legacies)

Why did this happen?

- The DNS is a little protocol, how did this mushroom into something so big?
- Web Browsers
 - "Completion of names" added .com first
- Trademarks
 - Corporations wanted their names "straight up" in DNS

What happened next?

- The creation of Internet Corporation for Assigned Names and Numbers (ICANN)
- Establishment of "alternate" roots

ICANN

- A result of a lot of lawyering
- Mission is to represent the world's needs of the DNS
 - Some success
 - And a lot of hot air

What has ICANN done?

- Created a domain name market
- Introduced competition in names
- Brought artificially high name costs down
- But, has it been a success?

The root zone

- The root zone, the contents of the DNS root are managed by a group of
 - ICANN
 - US Government
 - Verisign
- The US role is contested by many folks
- Why?

Why the US has its hands on

- The US government did pay for a lot of original development of the Internet, but that's water under the bridge
- The US government depends on the Internet for its performance, earlier than it should have been

Why folk want the USG out

- Other folks (nations, UN, ITU, citizens) have also come to rely on the Internet
- They don't implicitly trust the USG
- The debate - how much control does ICANN get?

ICANN's dilemma

- They are busy trying establish an organization over the "volcanic island"
- They are trying to be prepared for the responsibility
- The USG won't hand over it's role until ICANN meets some level of performance (what?)

Where ICANN is stalled

- It has so many business and legal issues it loses the voice of the people that attend to it
- Issues go unanswered
- A lot of it's power is limited by contracts
- ICANN is not capitalizing on the promise

Alternate Roots

- Some people dispute ICANN's authority and that of the USG
- They want to have their own DNS
- Problem is, other DNS versions don't have all the same names and data and don't have the customer base
- Some see these people as "crooks" for having alternate roots

The morale of this story

- DNS is a shared database with a large data space
- Who is in charge of it is in dispute
- Unless you ask someone who thinks they are the one
- With such an environment, trying to engineer is hard (what are the requirements?)

Technical DNS

- The DNS was defined in the late 80's by programmers and it morphed a bit through the 90's
- The basic protocol was extended a few times without fanfare or incident
- But then came security...

The DNS protocol

- The DNS query-response protocol is based on a unreliable substrate called the User Datagram Protocol
 - UDP is "best effort" - you're lucky if it gets through
 - *"Freedom"'s just another word for nothin' left to lose, and nothin' ain't worth nothin' but it's free - Me and Bobby McGee*

DNS on UDP

- DNS had to have five features to survive UDP
 - Extra sources/copies of data
 - Retry when there is no response
 - Compact messages
 - Simple (to restart) components
 - Flexibility in answering questions
- These make security hard - why?

Extra sources/copies of data

- When there are copies of data each has to be secured
- Can't count on one locked up version
- Need to "watermark" the copies to prove authenticity

Retry when there is no response

- Not so much a problem for security per se
- But there is a side effect, a related problem
 - DNS can be used to fill the network with packets, trusted packets

Compact messages

- Adding security data will make messages grow in size
- Significantly bigger - more than double or triple
- Also, security data may require more messages

Simple (to restart) components

- Security is a complicating factor, it makes algorithms more involved

Flexibility in answering questions

- Flexibility is the worst thing for security
- Security's goal is to disallow certain actions
- It's often hard to distinguish between dangerous actions and necessary actions

What else about DNS makes security hard?

- Scale
 - Delegation of management means coordinating a lot of people
 - High query activity means security has to be fast

DNS security goals

- Data Integrity
- Source Authenticity
- Authorization
- Message Integrity

Data Integrity

- Does the data received match completely the data sent?
 - Was data ripped out?
 - Was data added?
 - Was data completely fabricated?
- This does not mean that data sent was received!

The danger

- Without integrity
 - Was an entire answer received?
 - Was the received answer augmented?
- I could remove one source of data and not be able to remove another, so I can de-list the latter source in answers

Source Authenticity

- Did the data received come from the appropriate source?
 - Did someone forge the data?

The danger

- DNS responses are accepted "first come first used"
- If I "forge" an answer and return it before the true source, I win
- This can be used to send traffic to attack sites, like changing a highway sign in a road race, sending competitors the wrong way

Authorization

- Does the querier / requestor have permission to have an action performed?
 - Should the query be answered?
 - Should the operation be allowed?

The danger

- If unauthorized changes are allowed, you've lost control
- "Anonymous" dynamic updates can allow your webserver addresses to be moved without you noticing

Message Integrity

- Was the message received the one that was sent?
 - Did someone change the header or trailer?
 - Was this message a replay of an earlier one?

The danger

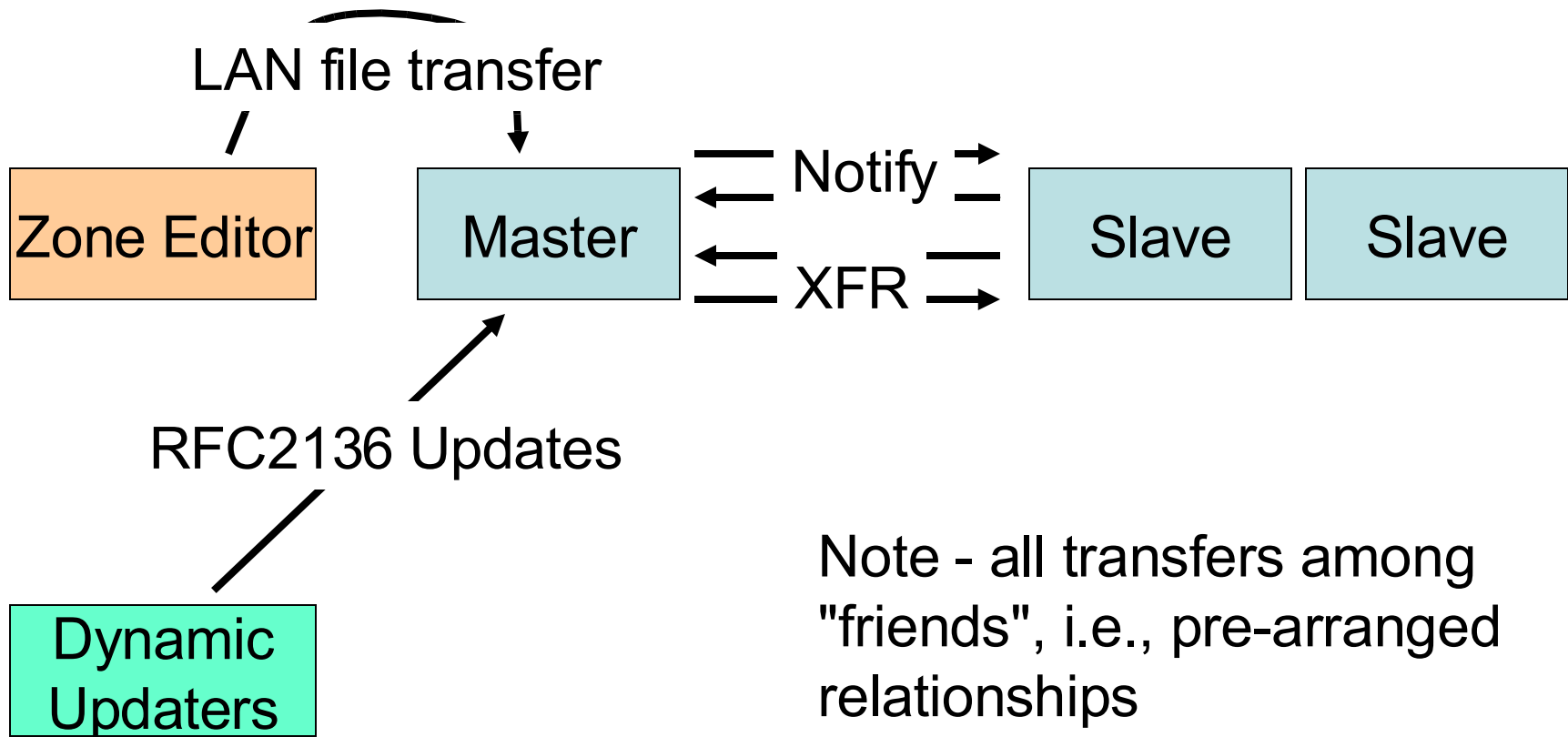
- Missing any part of a message or having something change can alter how the message is processed
- If security data is altered, a system becomes vulnerable

Retrofitting Security

- Start with a look at the protocol flows
- Not worried about machine access, process integrity, buffer overflows (Somebody else's problem!)
- Look at
 - DNS data entry and replication
 - DNS lookup

The DNS Machine

Part 1 - Entry and Replication



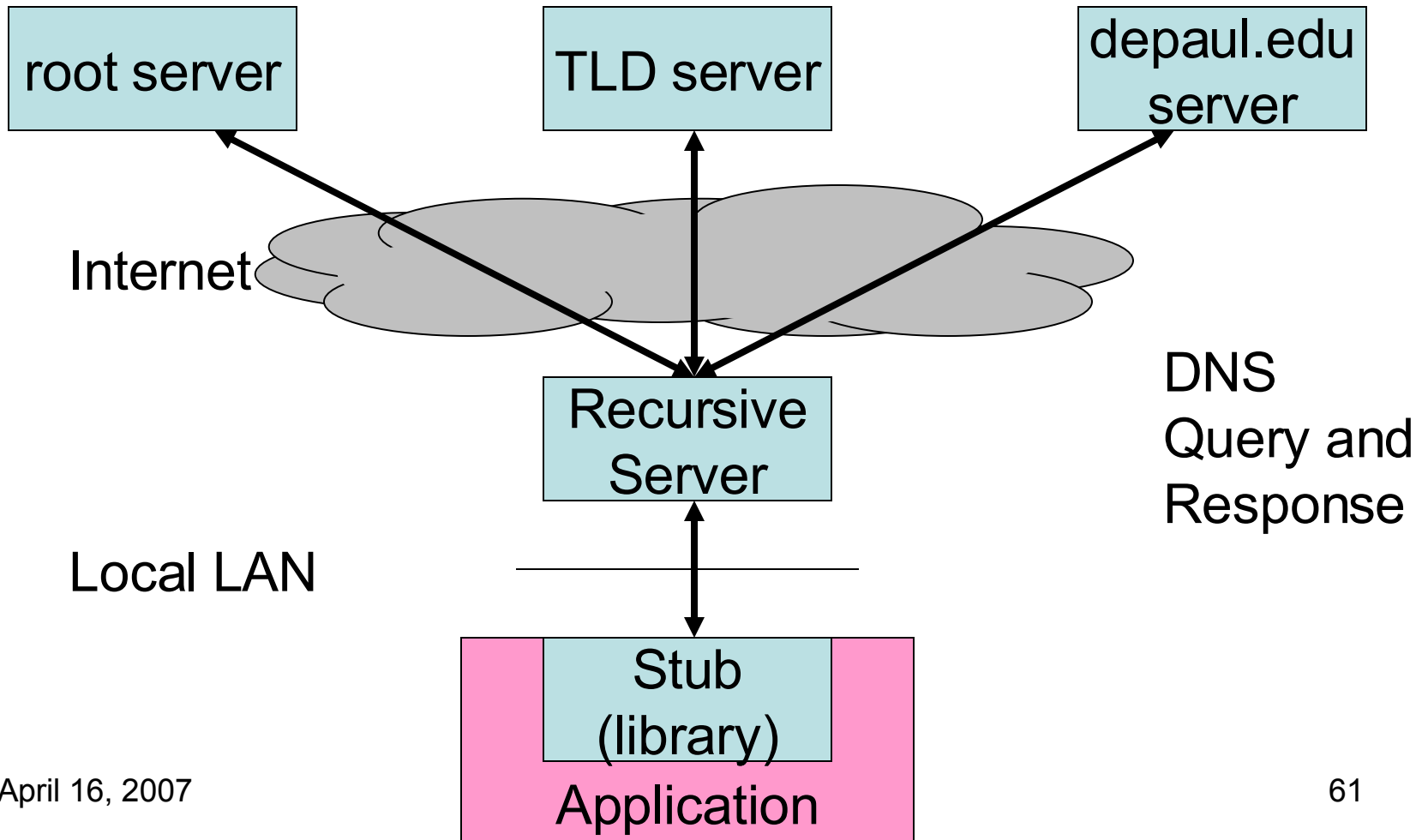
Note - all transfers among "friends", i.e., pre-arranged relationships

Strategy

- Use pre-arrangements to initiate security
- Message integrity for "safety" of messages
- Authorization (policy) to make changes
- Source authentication and data integrity for zone file

The DNS Machine

Part 2 - Queries



Strategy

- For Stub to Recursive use pre-arranged security
 - Same as previous
- From Recursive to Internet servers
 - Need a "public" approach
 - Need to chain trust about root to all others
 - DNS Security Extensions

Chaining

- Chaining trust (transitive trust) is hard
- In DNS, would like to trust just the root and have the root give evidence the next level can be trusted
 - Can we configure something about the root and start there
 - Can we get all DNS admins to do their part in extending trust?

Securing DNS

- The process to accomplish what I have described has taken 10-15 years
- My first exposure was in 1994, my first coding was in 1996, my first "how do we make this happen" meeting was in 1998, my first public workshop was in 1999...my most recent engineering workshop was in 2006

It's technical, what's gone wrong?

- It wasn't just technical, it is a problem for operations
 - Security egg-heads designed a secure but unworkable solution
 - That's been overcome for the most part
- The problem has changed
 - A cheaper solution was found to the bulk of the problem (but not all)

And then there's non-tech

- Non-technical reasons have come to dominate
- Remember ICANN and the root?
- Recall that the security needs to chain from the root?
- We have gridlock here

To sign or not to sign

- Ironic - people are willing to trust an arrangement to fill the root zone
- People are not willing to let the same arrangement sign or secure the root zone
 - I don't understand that
- Hence gridlock

It's not only the lawyers' fault

- Engineers caused problems too
- Securing a system with only security knowledge is a problem
 - A thorough knowledge of the underlying, unsecured, system is a must
- Security knowledge is important, or an extension will be haphazard

Shifting from data to transport

- Up to now, we looked at the protocol and saw some issues of securing the data in DNS
- Now we can look at security issues related to the transport of the data
 - Focus shifts to the fact that each DNS message takes up network capacity

What about network floods?

- We have looked only at the protocol design for security issues
- But the DNS system has other security risks
 - The use of the traffic to disrupt the network
 - The use of traffic to disrupt access to DNS

UDP problems

- UDP as a transport for DNS has benefits
 - lightweight and fast
- But UDP can be used for bad
 - Easy to forge
 - Easy to flood
 - "Fire and forget"

Packets behaving badly

- Denial of Service
 - Network level attack - clogs routers
 - Host level attack - clogs a host
 - Application level attack - kills a process
 - Distributed DoS means that many machines are talking needlessly
- Bad reactions to DoS

Bad reactionary security

- Sometimes a bad defense is more dangerous than no defense
 - Firewalls expect certain kinds of DNS traffic
 - May throw out good with bad

Time to sum up

- DNS security can begin with its data, but there are non-technical issues with who owns the root
- DNS transport is troublesome because of its simple nature
- Security support around DNS can hurt DNS more than help