VoIP Peering: Why and How

Jon Peterson

NANOG 46

June 2009

Early History of VoIP

- Experiments began in the early 1970s
  - RFC741, “Network Voice Protocol”
    - “secure, high-quality, low-bandwidth, real-time, full-duplex (two-way) digital voice communications”
- Why didn’t it catch on?
  - Nodes on the Internet were mainframes
  - As more personal nodes emerged, they lacked audio capabilities
  - Network itself not optimized for constant availability and low-latency delivery
- These conditions relaxed by the early 1990s for academia (workstations)
- By the mid 1990s, consumer computers had the necessary connections and audio capabilities
  - At the time, it seemed all preconditions were met…

Internet Telephony & Presence Browsing (1997)

Recent evolution of telephony

- Mobility fundamentally changed the telephone
  - Between 1987 and 1997, number of US mobiles increases 50x
- Truly “personal telephones”
  - Telephones tied to a person rather than a space
    - “Private” landline telephones live in private spaces
- Increased consumer tolerance for:
  - Dropped calls (dead spots)
  - Lousy codecs
- Telephones and text messaging
  - Never a landline feature, welcome to SMS
  - Notion that there’s more than one thing you can “do” with a telephone number
    - More than just “call it and see what happens”

The Diversity of VoIP Today

- VoIP encompasses:
  - Skype-like peer-to-peer with greenfield identifiers
  - Internet telephony clients calling from PC to phone
  - Primary line replacement like Comcast Digital Voice
  - Enterprise IP PBXs connecting to the PSTN
  - PSTN-IP-PSTN IXC replacement a la Level3
  - 3GPP Wireless VoIP
  - IP Conference bridges for Internet gaming

(and that's not including streaming applications)

Application-layer Peering

- Not the same thing as peering at layer 3
- Establishment of bilateral (or multilateral) pre-associations between service providers
  - Specific to a particular application; could have different relationships for different apps
  - The association is a matter of policy, though policy has various operational ramifications
  - Users by definition can't "peer" unless they act as service providers
  - Contrasts with an open service model where no pre-association is expected

Historical Peering of Applications

- Email
  - Domain-level black/whitelist
  - Dedicated lines or peering points linking very high-volume providers
    - Also mobile providers
  - More recently, DKIM
- IRC
  - Service is run by the service providers, and servers establish bilateral relaying agreements with one another to create a network
- Similar practices in the web (caching, content distribution) and in usenet

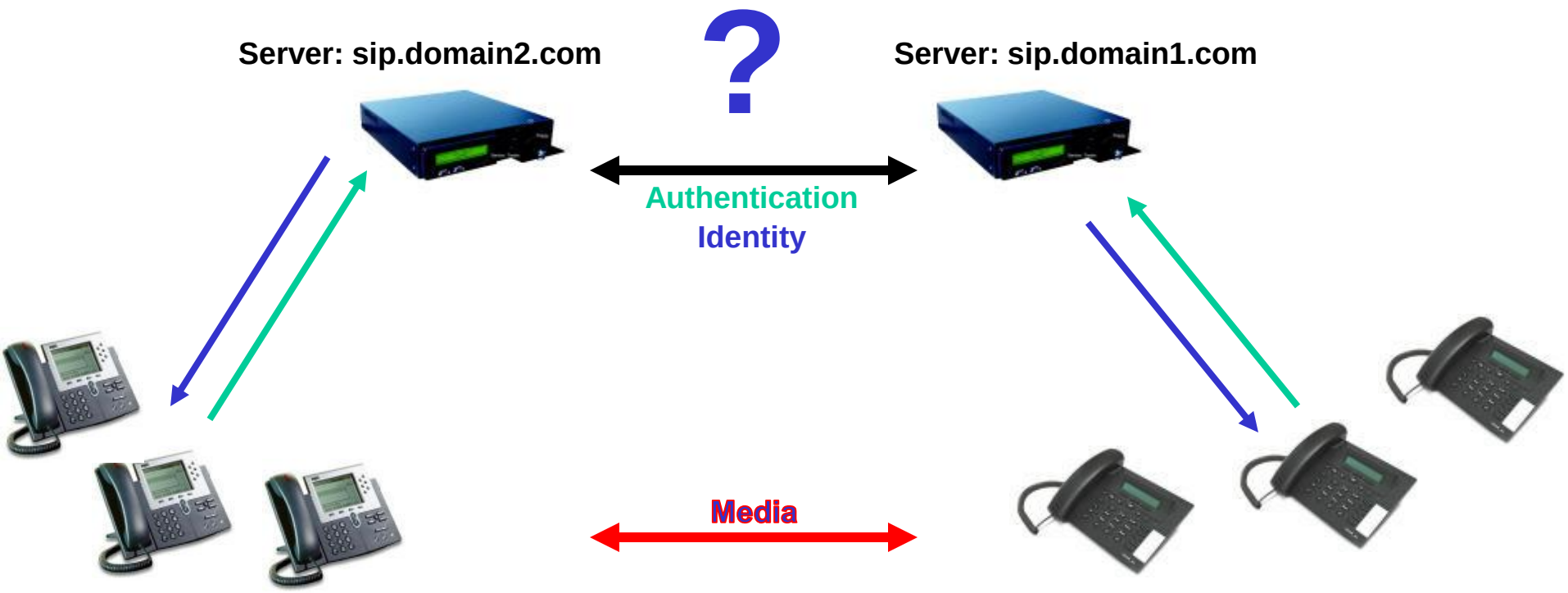
Telephony on the Internet

- **On the Internet, telephony is an application**
  - Not necessarily a service, no service needs to be provided
  - A developer can always just write and distribute a new VoIP application
  - Protocols were designed with these assumptions
- **The IETF has built a suite of real-time communication tools**
  - These tools can be used to instantiate telephony
  - Includes SIP, SDP, RTP and related protocols
- **Where there are services, however, there may be peering**
  - The IETF SPEERMINT WG studies these requirements
    - The IETF DRINKS WG designs the necessary provisioning
  - Why peer if the application doesn't require services?

VoIP as a Service: Why?

- Some sort of registrar/proxy function is essential
  - Manages your address of record, translate to your current endpoint
  - May also keep track of utility endpoints like voicemail while you are offiline
- Gateways to the PSTN
- NAT Traversal
  - ICE (STUN/TURN) or just an ALG
- Security
  - Mere firewalling
  - Identity management
  - To keep parity with the PSTN
- Once you have multiple service providers, how do they interact with each other?

How Should Domains Exchange Traffic?



VoIP Peering: Between Services

- Call routing
  - To another network (directly or no)
  - To the PSTN for telephone numbers, at worst
- Path assurance
  - Getting media through firewalls and NATs
  - Guaranteeing quality
- Accounting
  - Knowing who did what (for fraud/abuse)
  - Network capacity measurement and planning
- Generally aspires to PSTN-level security

Is Peering Right for Me?

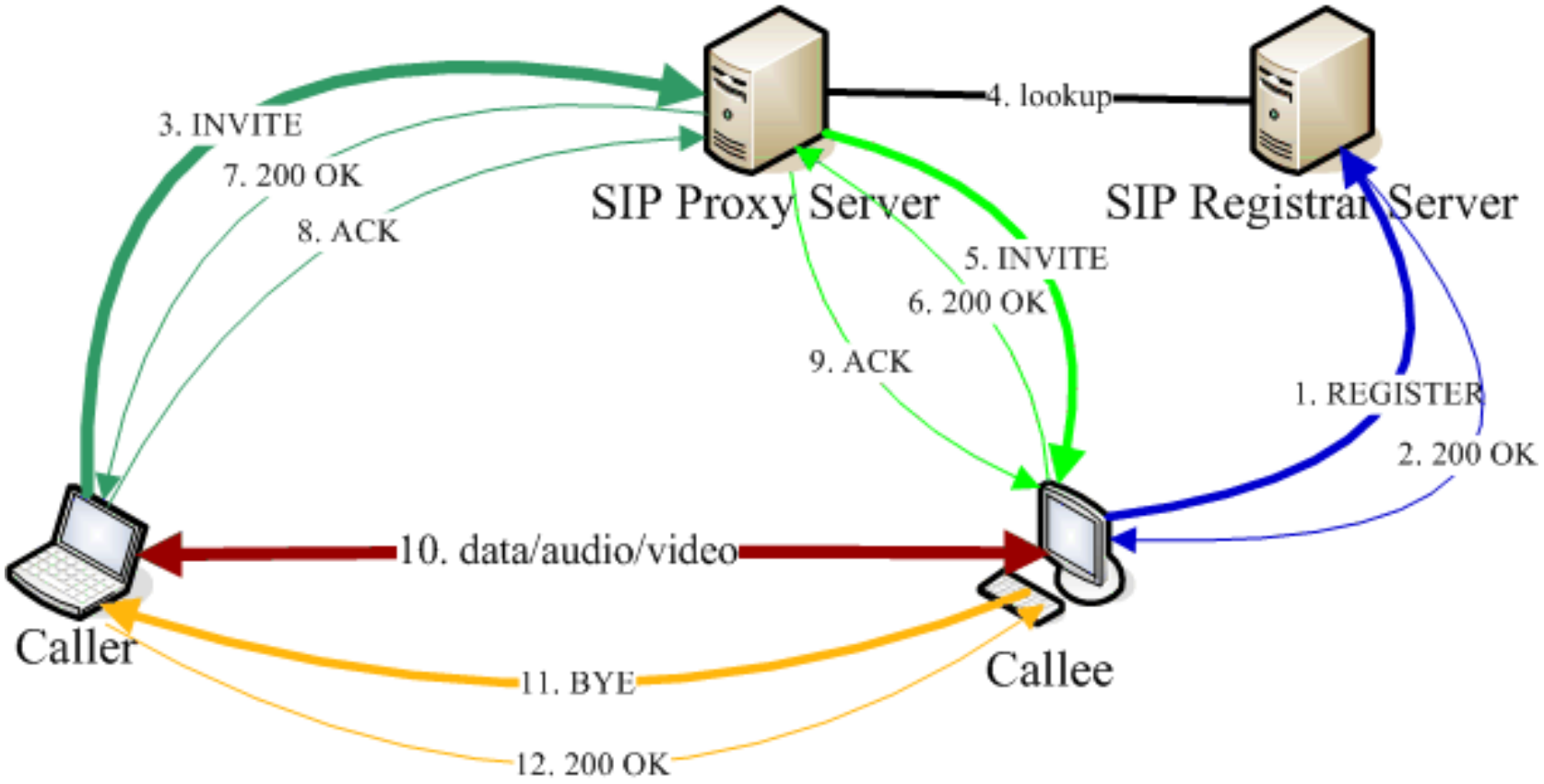
- Not all VoIP applications even benefit from peering
  - Conference bridges for Internet gaming, for example
  - Tried to define “peering” in a way still relevant to minimal greenfield, however
- You peer when your voice application benefits from Metcalfe’s Law
  - In other networks are more endpoints
- Cost-avoidance, that is, avoiding a PSTN routing leg, is a primary driver
- Many see VoIP peering as a potential successor to the PSTN

Also bear in mind…

- SIP is not just for VoIP
  - By resolving interdomain traffic policies for SIP, we also address:
    - SIP-based instant messaging, videoconferencing, etc
- And, VoIP is not just SIP
  - However, virtually all VoIP has a similar architecture
    - Registration, proxies, media relays for NATs, etc
  - Even many proprietary VoIP systems use SIP as a *lingua franca* to peer with the outside world
  - The lessons we learn from studying SIP peering are likely to be more broadly applicable

Call Routing

A Reference SIP Call Flow

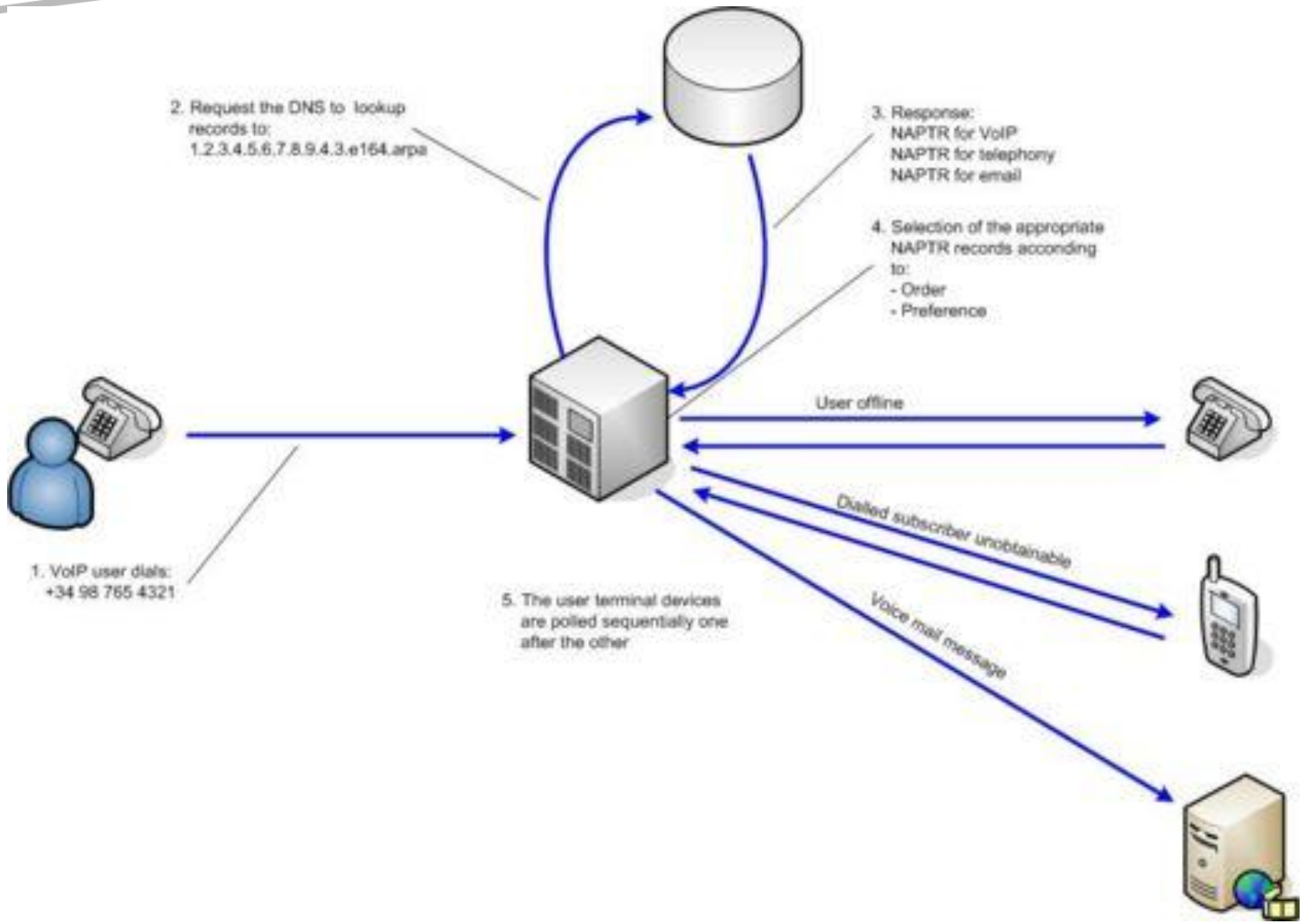


http://www.javaworld.com/javaworld/jw-06-2006/images/jw-0619-sip2.gif

Naming and Addressing

- SIP can use telephone numbers and email-style URIs
- Names and addresses
  - A name is a pointer, an address is a location
    - www.stanford.edu points to many IP addresses
- A telephone number is a name
  - Consider freephone numbers, or number portability
    - Find Me/Follow Me, voice mail
    - Translating it to an address in the PSTN is complicated
    - Even worse on the Internet (ENUM)
- Why not just use URIs?
  - Telephone numbers are deeply entrenched
    - Non-linguistic and thus international
    - Necessary for PSTN interworking
    - IP telephones have numeric buttons (even Net2Phone)

ENUM: Mapping Telephone Numbers to URIs



Where to send an INVITE?

- SPEERMINT divides this into two questions:
  - What is the domain to which the INVITE should be sent? (the “LUF” or “look-up function”)
  - Which element in that domain handles INVITEs from outside? (the “LRF” or “location routing function”)
- These two questions may be answered by different services
  - For example:
    - LUF: ENUM translates telephone number +15714345400 to sip:+15174345400@neustar.biz
    - LRF: Via RFC3263 (DNS), the proxy for neustar.biz = sip.neustar.biz

Peering Architectures

- Assume for the moment that layer 3 interconnection is taken for granted
- Direct versus indirect
  - Do two networks connect through an intervening network (i.e. application-layer processing by an intermediate administrative domain) or directly?
- Static versus on-demand
  - Do two networks share an agreement to exchange traffic, or do they lack pre-association and connect on-demand?
- Federations
  - Essentially static, indirect multilateral peering

Deployment Implications

- The edge elements operated by the service providers must have relevant LUF/LRF capabilities
  - LUF and/or LRF can be local or queried in the network
  - Service provider policy, typically, must groom the results of both
    - This is one of the reasons public ENUM has not been a great success, but instead private services flourish
  - The best way to reach a particular peer might not equal the default DNS response, for example
- VoIP service providers are reluctant to expose internal topology
  - Border elements shield service provider networks
    - In SPEERMINT, SBE (Signaling-path Border Element)
  - Provider policy also plays into admissions decisions

Path Assurance

Old News

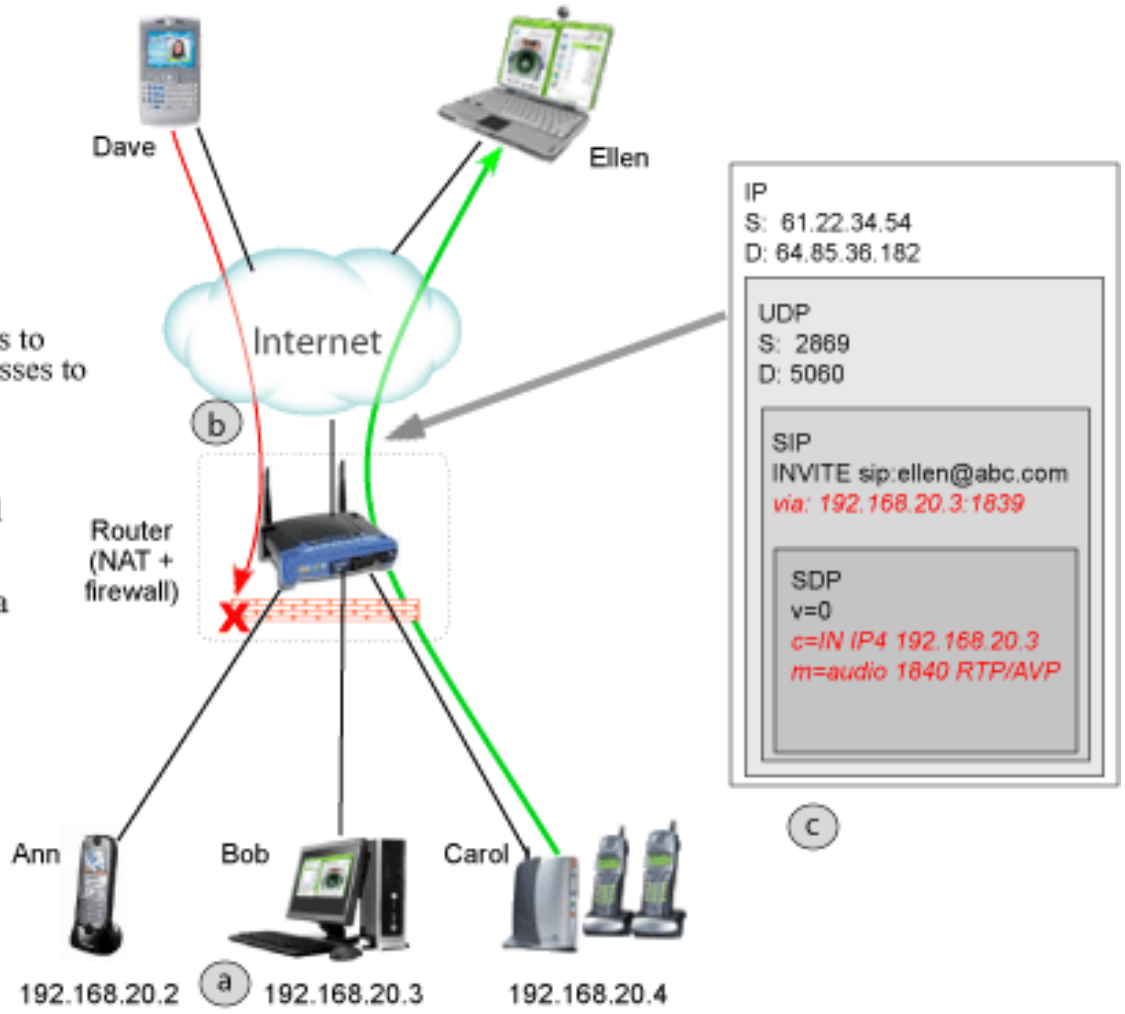
- Many VoIP services have endpoints behind NATs
  - Consumers behind a home NAT
  - Business phones behind enterprise NATs
  - Even carrier networks deployed in private (or IPv6) address space
- Famously, protocols like SIP interact poorly with NATs
  - SIP carries SDP, which contains the IP addresses and ports for the media session

NAT Traversal Woes

(a) NATs provide private IP addresses to local computers and map these addresses to public IP for Internet access.

(b) Firewalls protect devices from external threats by filtering uninvited IP packets. Carol can call Ellen, allowing Ellen's response to reach Carol. But user Dave cannot initiate a call to user Carol.

(c) Problem of making VoIP calls between users behind NATs



www.nattraversal.com/nattraversal2.htm

Address Fixing

- The network/transport layer obstacles can't be ignored and aren't going away
  - Aside from consumer NATs, now LSNs, to say nothing of v6 transition devices
- A "media relay" is therefore a necessary part of the peering architecture
  - Not required in all cases, but required to cover all
  - Relays may be implemented as application-layer gateways (typical SBC model)
  - Also TURN, a component of the ICE architecture along with STUN, may relay media

Other Path Responsibilities

- Once media is going through a box, it is easy to do other things with it…
- “Media Steering”, to anchor traffic through a specific set of application-layer relays
  - Overriding layer 3 packet routing, thus controversial
  - Especially problematic in indirect peering
- Various forms of monitoring and logging
  - Commonly for fiduciary services or for CALEA compliance
- DPI (deep packet inspection) to enforce various policies
  - Some service providers restrict codecs, for example
- Resource reservation for quality of service
  - May include opening firewall pinholes

What kind of media relay function do you need?

- Not all VoIP services have the same media relay needs
  - Peer-to-peer file-sharing teaches many lessons about pragmatic comprises
  - Topologically, some peers may require less path assurance than others
- Media relays must be designated in SIP signaling
  - Either by the endpoints, or unilaterally by an intermediary
- Today, support for ICE is uncommon and session border controllers (SBCs) are common
  - Ultimately, the SBC approach works, though with some caveats about security and extensibility
    - At a high level, smarts in the network can threaten innovation in the endpoints
  - Lightweight functionality here goes a long way

Accounting

First: “Know who did what”

- A deceptively simple capability
- In SIP and related protocols, this problem is largely cast as an “identity” problem today
  - Some solutions, but not a great deal of consensus
- In the PSTN, transitive trust is the norm
  - Walled garden, closed network (static, indirect)
  - The closer a VoIP environment is to that, the more it can rely on those principles
  - Closing networks, however, limits the addressable user base
  - So how can we do better?

What are the identity threats facing VoIP?

- Caller-ID falsification
  - Gotten a fair amount of press in the past, high profile issue
  - Fraudulent caller-ID might ghettoize VoIP, interfere with some 911 solutions, etc
- SPIT (Spam for Internet Telephony)
  - VoIP spam (repeating announcements, etc) is no doubt a tantalizing prospect for some
    - Use of SIP for IM is also susceptible to spam
  - Without identity, there can be no accountability
- Customer identification/authorization for services
  - Authorization policies implicitly assume the ability to identify senders

SIP Security Fundamentals

- **SIP is difficult to secure**
  - Proxies make end-to-end message confidentiality problematic
    - They need to read messages to route them
  - Also, proxies add and modify certain headers
    - End-to-end integrity also therefore a problem
- **Caller-ID (identity) is a very desirable security service**
  - In the PSTN, Caller-ID mostly works through transitive trust
    - (i.e., it doesn't work)
  - In the SIP world, user agents have credentials in their local realm
    - Much like HTTP username/password
    - Passwords, however, require pre-association
    - Telephony needs to work even when there is no pre-association
- **The result: a hodgepodge**
  - Digest authentication, TLS, possibly S/MIME, other situational mechanisms

Security mechanisms in baseline RFC3261 SIP

- **Digest**
  - **Based on HTTP Digest (Basic has been deprecated)**
  - **Can be used to derive authentication properties (based on shared secrets) and some integrity properties**
  - **Useful for the SIP registration function (likely to share a secret with the registrar)**
- **TLS**
  - **Which is of course the new name for SSL**
  - **Gets canonical properties: integrity, confidentiality, mutual authN, replay protection)**
  - **Authentication requires certificates**
- **S/MIME**
  - **Provides security for the bodies of SIP messages**
  - **Various ways to use this to secure header information as well**
  - **In practice, S/MIME is unused in SIP implementations**

Survey of SIP Identity Solutions

- Domain-level transitivity (secure connections)
  - Bilateral
    - Domains connect with a VPN, or mutual TLS
    - Keep track of the source of traffic, use that to make an authorization decision
    - SBCs are common tools of this approach
  - Multilateral ‘clearing house’ approach also possible
    - GRX/CRX
- Federated (secure assertions)
  - RFC3325 approach, everyone trusts P-Asserted-Identity within the federation
    - In this case, you don’t even know what domain it came from, just that it was within the federation
  - Identity (RFC4474)
  - Maybe web services identity (SAML, OAUTH, etc)

Authentication Service

INVITE (password)

sip:auth.domain1.net

INVITE (identity)

Auth.example.com checks the credentials, and creates Identity and Identity-Info headers, adds them to requests

Auth.example.com forwards the request to the destination – they compare the From header to the asserted identity

sip:alice@domain1.net

sip:bob@domain2.com

A Four-Layer Security Model of SIP

- **Transaction Security Layer**
  - Use of Digest and TLS to secure SIP transactions and authenticate user agents to their domain
- **Identity Layer**
  - Identity Signature over certain headers and the SDP, securing media descriptors like codecs and the network/transport address
  - Signature is authorized based on authentication from Digest
- **Media Keying Layer**
  - Presence of keys or key fingerprints in SDP
  - Integrity for keying material derives from Identity layer
- **Media Security Layer**
  - Use of SRTP to provide confidentiality and integrity of RTP media
    - Keyed according to the signaling in SDP

- **Each layer depends on the one above it**
  - **The goal is to let the application know if the media is secure**

Identity is key to much of SIP's operation

- Presence, for example, is based on authorization
  - The "buddylist" being the most familiar tool
  - Share presence information only with trusted parties
    - May give others nothing or even lie to them
  - Fine granularity in sharing presence information
    - Some may get a subset of my presence
- Authorization requires identity
  - Who is making a request?
    - How do I know the requestor is who they claim to be?
- Presence also requires identifiers
  - A name to subscribe to
- Identity profile information may include presence

Known problems with the RC4474 model

- Problems with the overall Identity model
  - E.164 numbers don't work well with Identity
    - Signatures rely on domain portion of URIs
  - "Aggressive" intermediaries change signaling in ways that break Identity signatures
- In those environments, however, Identity might not be the best approach
  - Existing solutions do include both 4474 and 3325
    - P-Asserted-Identity is often workable in the sorts of environments where "aggressive" intermediaries live
  - These environments are, however, common

“Aggressive” Intermediaries and signaling

- **Intermediaries do not restrict themselves to the RFC3261 “amdr” rules of proxies (used by RFC4474)**
  - However, scope of intermediary agency must have practical limits
  - Otherwise, there is no way to differentiate legitimate actions from attacks and no scope for protecting SIP signaling
- **UAs implicitly authorize some intermediary alterations and not others**
  - It seems clear that UAs do not, for example, authorize intermediary changes to the key fingerprints in SDP
- **Today, all this remains poorly understood**
  - We would need the real “amdr” before we get into solutioneering
  - That requires, essentially, some formalization of SBCs

Can ENUM help with E.164 number identity?

- **ENUM might be used to look up the domain responsible for a telephone number**
  - NAPTR record containing pointer to Identity-Info URL
  - Potentially that domain could be used for Identity signatures
- **Making this available as a service to verifiers seems attractive**
  - Some incentive to deploy public ENUM, even
- **Difficulties appear insurmountable, though**
  - Establishing ownership of a number is difficult
    - Binding a particular call to an owner seems to be impossible
  - Also, service providers might be uncomfortable disclosing their ownership of numbers

Solving Accountability

- Most federation-style peering is effectively closed
  - Simple transitivity with RFC3325 PAID probably suffices
- Greenfield deployments without telephone numbers are also easier to solve for
- In some other places, we still don't have good answers:
  - Cases where there might be PSTN interworking
  - Cases where (helpful) intermediaries modify SIP and SDP when the Identity header has been used
- Service providers may make decisions based on the originating domain rather than a more granular identity
- Caller-ID may be no worse than the PSTN's anyway

Matching the PSTN

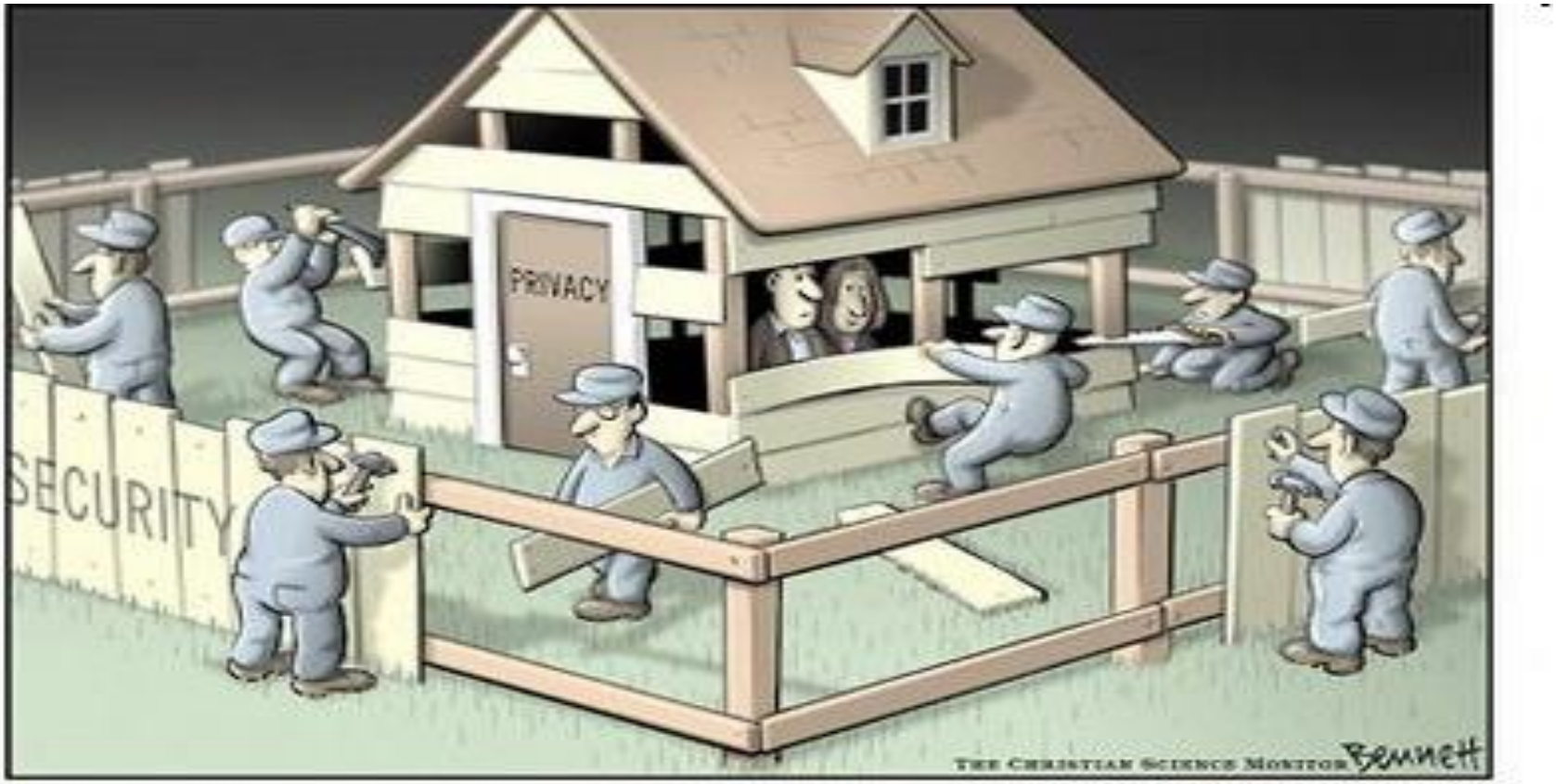
Security of the PSTN

- For the customer:
  - **Protection from eavesdropping**
    - Closed network
  - **Authentication of call source**
    - Caller ID
  - **Calls cannot be diverted**
    - Trust the network
  - **Uninterruptable service**
- For the operator:
  - **Avoid toll fraud**
    - CCIS
  - **Prevent vandalism/DoS**
    - CCIS
  - **Honor regulatory constraints**
    - CALEA, GETS

The Purpose of Security

- Security mitigates attacks and risks
  - In the absence of specific threats, there is no need for security
  - To be effective, security must be routine and will almost always be employed in the absence of attacks
- Some attacks can be thwarted, others merely detected
  - It is the responsibility of applications and users to behave properly when attacks occur
    - Users are notoriously unreliable; ultimately user responsiveness limits the scope of security
- Sometimes we use security to increase our privacy, prevent impersonation and tampering with messages, but…

Two Properties at Odds



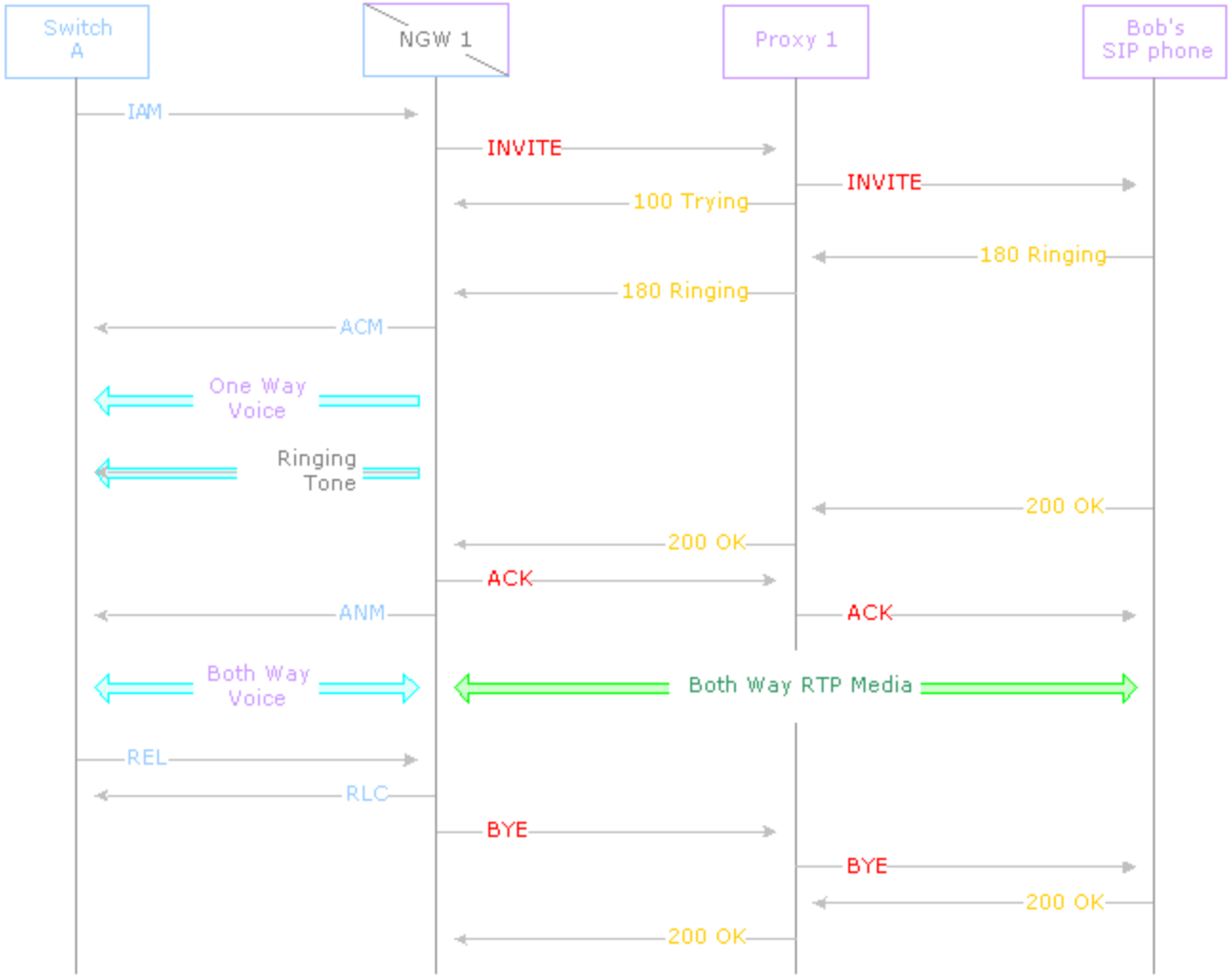
But how secure is the PSTN (for consumers)?

- Caller ID isn't really a security mechanism
  - It works as long as no one is trying to fool you
  - Telemarketers routinely flout it, either withholding IDs or sending false ones via Q.931
  - Widespread Internet apps also spoof Caller ID
- Eavesdropping is also problematic
  - Again, your call is private as long as you're trying to keep it private from people who can't eavesdrop
  - Recourses are legal rather than technical
    - Acoustic coupler encryption, circa 1980
- Both are provided at the discretion of the operator

Walling a Garden

- The PSTN's walled garden is a private SS7 network
  - Though the exclusivity of the club is much diminished lately
- For static peering, VPN architectures are a popular choice
  - Extending the walls around your lawn into a private commons
  - IPsec gateways to extend enterprise networks across stretches of public Internet
- For dynamic peering, TLS and the standard RFC3261 procedures
  - Requires minimal pre-association – just a common root CA
- For very high bandwidth videoconference, even layer 1 private networks may be necessary

PSTN Interworking (or How to Join the Club)



Emergency Services

- Several regulatory requirements make PSTN interworking complicated
  - What if someone dials 911?
- For now, we need to get the call to the PSTN
  - Need to reach right Public Safety Access Point (PSAP)
    - This requires coarse geographical location of the caller
  - Getting geolocation associated with Internet devices is tricky
    - VPNs and so on lead to trivial mistakes
    - Easier for mobile devices, or those with embedded GPS
  - SIP messages can now carry geolocation (RFC4119)
    - Mechanisms for mapping geolocation to PSAPs exist
  - Security implications, and potential for abuse, are huge
- Future requirement: to access IP-enabled PSAPs

Will Peering Architectures Predominate?

- Unclear, today
- From an adoption perspective, peer-to-peer VoIP dominates
  - 1B (!) downloads of Skype, Sep 28 2008
  - 13M user concurrency on Skype, Sep 15 2008
  - Largest market-share first culture provider still <10M
- However, commercial VoIP interest lies with peering
  - Traditional telephony vendors make SPEERMINT-style devices
    - Traditional telephone service providers deploy them in closed environments,
  - Even Skype's business is indeed gatewaying via peers
  - However, traditional telephony regulatory concerns limit success

History Lessons

- Dominance comes from standardization
  - Though we should not accept this unquestioned…
- Standards for email came slow
  - Islands were the norm for a decade or more
  - Standardization linked the islands but did not pervade them
  - The exchange of mail (interdomain) is settlement-free
    - Business model is largely in enterprise sales of servers and clients
- What would it mean to replace the PSTN?
  - PSTN serves many purposes other than delivering personal voice communications
    - Fax and modems, TTY, but also calls for commercial or other purposes that today the web does best
  - How much standardization is thus needed, given this diversity?

The Moral of the Story

- The components in peering address real problems
  - NAT traversal, routing to E.164 numbers on the Internet, providing trustable Caller-ID
    - Peering also addresses many problems that become low-hanging fruit when you've addressed the above
  - There is no magic bullet single solution to peering
    - Answers come *a la carte*, not *prix fix*
    - Particular domains may not need them all to peer
- If you specify peering so that it does not rule out on-demand, direct cases, it will always have relevance