SIP Operation in the Public Internet

An Update on What Makes Running SIP a Challenge and What it Takes To Deal With It

Jiri Kuthan, `iptel.org`

sip:jiri@iptel.org

Outline

- Status update: where iptel.org's operational experience comes from and what works today
- Trouble-stack: things which do not fly yet
- Operational Practices
- Conclusions

Background

- iptel.org has been running SIP services on the public Internet since 2001. Users are able to pick an address username@iptel.org and a numerical alias.
- The infrastructure serves public subscribers as well as internal users with additional privileges (PSTN termination, voicemail).
- Services powered by open-source SIP server, SIP Express Router (ser).
- Increase in population size since introduction of Windows Messenger: free Microsoft SIP client with support for VoIP, video, instant messaging and collaborative applications.

Good News …

- Basic VoIP services work, so do complementary integrated services such as instant messaging, voicemail, etc.
  - *Commercial deployments exist, mostly offering PSTN termination: Vonage, deltathree, denwa, Packet 8*
  - *Trial services: FWD, PCH, WCOM, SIP Center*
  - *Tens of intranet deployment of SER reported, probably many more unknown*
- Billing machinery works too: Accounting easy, though not standardized.
- Numbering plans easy to maintain and they complement domain names well.

… Good News

- QoS mostly pleasant for broadband community:
  - *Links between iptel.org site and iptel.org user community have packet loss close to zero and RTT mostly bellow 150 ms, rarely above 200 ms.*
- SIP interoperability well established across mature implementations
- Interoperation with other technologies works too:
  - *Competition on the PSTN gateway market established*
  - *Gateway to Jabber instant messaging up and running*
  - *Commercial H.323 gateways exist*

Bad News

- Nightmare – NATs (…)
- Why I keep my PSTN black phone in my room's corner: Reliability (…)
- What Is It? Machines Do, Operators Don't … Scalability (…)
- End-devices still expensive
- Future issues: spam, denial of service attacks

NAT Traversal

- NATs popular because they conserve IP address space and help residential users to save money charged for IP addresses.
- Problem: SIP does not work over NATs without extra effort. Peer-to-peer applications' signaling gets broken by NATs: Receiver addresses announced in signaling are invalid out of NATted networks.
- Straight-forward solution: IPv6 – unclear when deployed if ever.
- There are many scenarios for which no single solution exists (they primarily differ in design properties of NATs – symmetric, app-aware, etc.)

Current NAT Traversal Practices …

- Application Layer Gateways (ALGs) – built-in application awareness in NATs.
  - Requires ownership of specialized software/hardware and takes app-expertise from router vendors (Intertex, PIX).
- Geeks' choice: Manual configuration of NAT translations
  - Requires ability of NATs, phones, and humans to configure static NAT translation. (Some have it.) If a phone has no SIP/NAT configuration support, an address-translator can be used.
- UPnP: Automated NAT control
  - Requires ownership of UPnP-enabled NATs and phones. NATs available today, phones rarely (Snom).

… Current NAT Traversal Practices

- STUN: Alignment of phones to NATs
  - Requires NAT-probing ability (STUN support) in end-devices and a simple STUN server. Implementations exist (snom, kphone).
  - Does not work over NATs implemented as “symmetric”.
  - Troubles if other party in other routing realm than STUN server.

  \+ Works even if NAT device not under user’s control.
- Relay: Each party maintains client-server communication
  - Introduces a single point of failure; media relay subject to serious scalability and reliability issues

  \+ Works over most NATs

NAT Practices: Overview

	ALG	STUN	UPnP	Manual	Relay
Works over ISP's NATs?	N/A	Ltd. (*)	N/A	N/A	Maybe
Symmetric NATs?	N/A	No	N/A	ok	Ltd.
Phone support needed?	No	Yes	Yes	Yes	Yes
NAT support needed?	Yes	Ltd. (*)	Yes	Ltd. (+)	No
Scalability	? (o)	Ok	Ok	Ok	poor ☒
User Effort	Small	Small	Small	Big ☒	Small

*… does not work for symmetric NATs

o … application-awareness affects scalability

+ … port translation must be configurable

NAT Traversal Scenarios

- There is no “one size fits it all” solution. All current practices suffer from many limitations.
- iptel.org observations for residential users behind NATs: Affordability wins: SIP-aware users relying on public SIP server use ALGs or STUN. First UPnP uses sighted.
- Our plan: hope for wider deployment of
  - STUN and STUN-friendly firewalls
  - ALGs
  - UPnP-enabled phones and NATs

Murphy's Law Holds

Everything can go wrong.

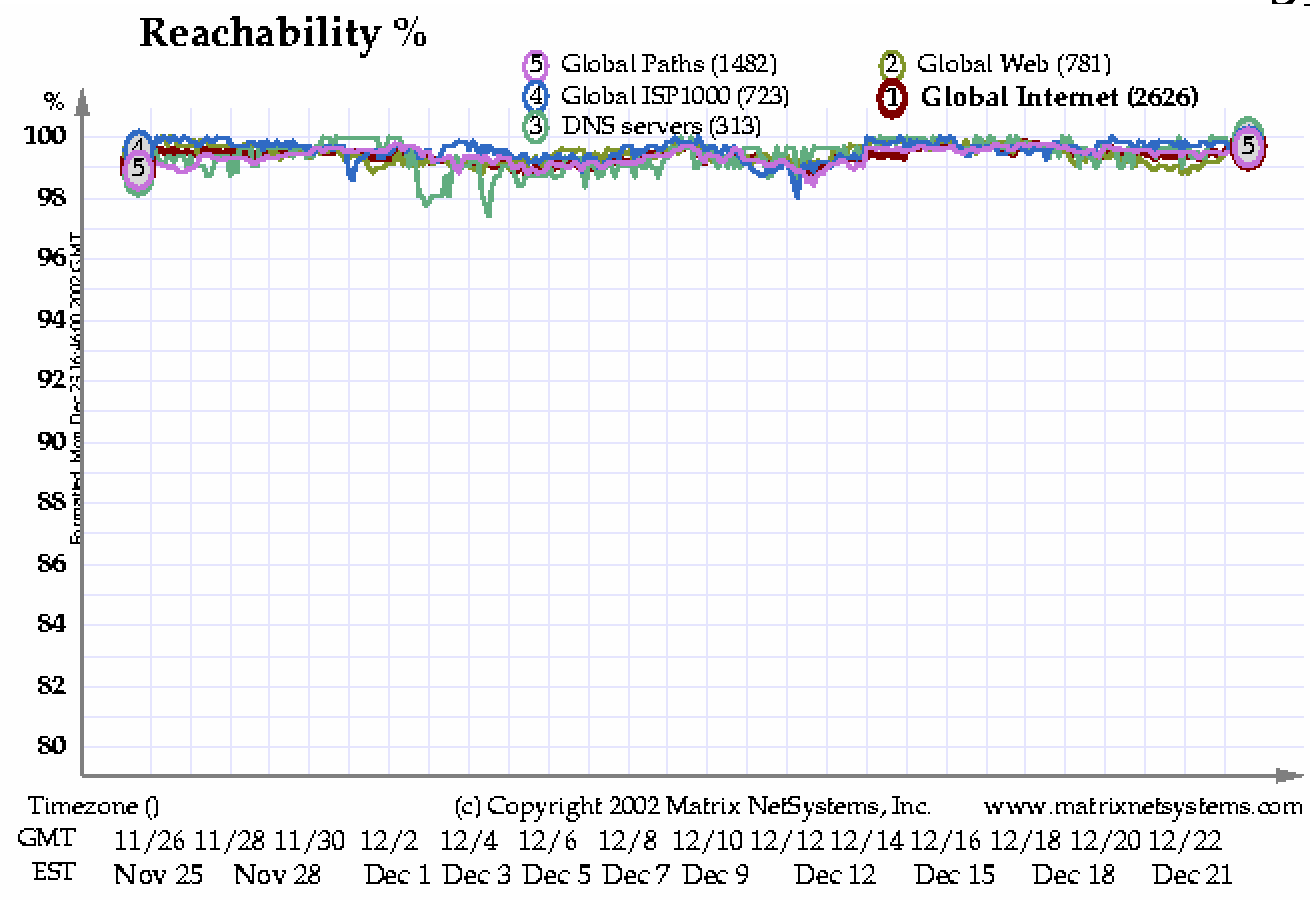
- Servers:
  - software/configuration upgrades
  - vulnerabilities
  - both SIP and supporting servers subject to failure: DNS, IP routing daemons
- Hosts:
  - power failures
  - hard-disk failures
- Networks:
  - line.
  - IP access

IP Availability: SLAs

- Industry averages for "Network Availability" SLAs are from 99.9% to 99.5% (an NRIC report)
- SLAs mostly exclude regular maintenance and always Acts of God
- Residential IP access rarely with SLAs

Availability (percent)	Actual Downtime (per year)
99.999	5 Minutes
99.9	9 Hours
99.5	1.8 Days

matrix.net's Reachability Statistics



- Minimum 98.69%
- Median 99.45%
- Maximum 99.84%
- Mean 99.40%

Wenyu Jang, Henning Schulzrinne: "Assessment of VoIP Service Availability in the Current Internet", in PAM 2003.

… 99.5%

Fail-over Issues

- Whatever the reason for a failure is, signaling needs to be available continuously. Most important components are:
- *Replication of user information*
  - Doable; using SIP gains better interoperability and avoids issues with database caches.
- *Making clients use backup infrastructure on failure*
  - SIP specification can do that (DNS/SRV) but today's SIP phones cannot (except one).

Fail-over Workarounds and Limitations

- IP Address Take-over: Make backup server grab primary's IP address when a failure detected
  - *Cannot be geographically dispersed, unless coupled with re-routing*
  - *Primary server needs to be disconnected*
- DNS Update: Update server's name with backup's IP Address
  - *DNS propagation may take too long, even if TTL=0 (which puts higher burden on clients)*
- Both methods rely on error detection which may be tricky – a pinging host may be distant from another client and have a different experience

Scalability Concerns

- New applications, like presence, are very talkative
  - *Presence status update frequent*
  - *Each update ventilated to multiple parties*
- Broken or misconfigured devices account for a fair load share; few of many real-world observations:
  - *Broken digest clients resend wrong credentials in an infinite loop → heavy flood*
  - *Mis-configured password: a phone attempted to re-register every ten minutes (factor 6) →2400 messages a day*
  - *Mis-configured Expires=30 (factor 120)*
- Replication, Boot avalanches, NAT refreshes

Achievable Scalability

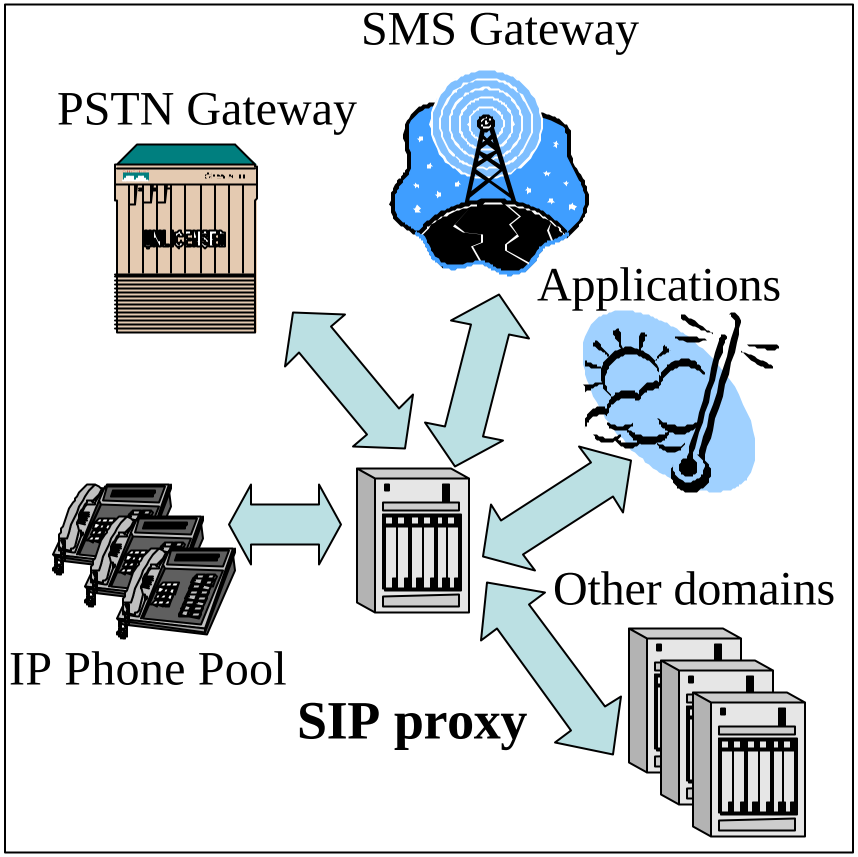
- Good news: well-designed SIP servers can cope with load in terms of thousands of calls per second (CPS)
  - *Example: lab-tuned version of SIP Express Router achieved transactional throughput in thousands of Calls Per Second on a dual-CPU PC – capacity needed by telephony signaling of Bay Area*
- Pending concern: denial of service attacks
  - *Example: hundreds of megabytes of RAM can be exhausted in tens of seconds with statefull processing*

Deployability

- Devices can be made scale, administrators not
- Well-known burdens:
  - Many boxes deployed consume many administrators.
    - **Network-building practice: Integrate signaling logic in as few boxes as redundancy strategy allows.**
  - Phones are not yet plug-and-play, particularly if behind NATs
    - **It is still phone vendors' turn.**
  - SIP routing good but not easy (…)

SIP Routing

- Benefit of SIP: Ability to link various service components together.
- The "glue" are signaling servers. Their primary capability is routing requests to appropriate services.



- Issues:
  – *Routing flexibility – how to determine right destination for a request*
  – *Troubleshooting when routing failures occur*

Routing Was Never Easy

- Request processing policy may be quite complex:
  - PSTN destinations require SIP servers to stay in the path for purpose of accounting and admission control.
  - Some destinations are reachable for anonymous callers whereas others take authentication and admission control.
  - Requests from originators known to support NAT traversal may receive different treatment.
  - Method-based routing – requests to PSTN are split by method between SMS and PSTN gateway.
  - Further factors include request's transport origin, address claimed in From header field, content of Contact, etc.
- **Operational observation: mighty tools for specification of routing policy are needed.**

Routing Language

- Our answer: routing language
- Features: conditional expressions may depend on any of previously mentioned factors; example:

```
/* free destinations, like Jiri’s mobile phone listed in an SQL table, or any
   local PBX numbers require no authentication */
if ( is_user_in("Request-URI", "free-pstn") | uri=~"sip:[79][0-9][0-9][0-9]@.*“ ) {
   log (“free call”); /* no admission control – let anyone call … */
} else { /* all other destinations require proper credentials */
    if (!proxy_authorize("iptel.org" /* realm */,"subscriber" /* table name *) {
      proxy_challenge(“iptel.org”, 0);
      break;
    }
    /* detailed admission control – long distance versus international, etc…*/
    if (uri=~"sip:0[1-9][0-9]+@.*") {
         if (!is_in_group("local")) {
             sl_send_reply("403", “Forbidden...");
...
```

SIP Routing: Troubleshooting

- SIP request can be routed along arbitrarily complex path
- Failures in numbering plans and SIP-routing in general difficult to locate without knowledge of:
  - Which Request URI caused an error
  - At which spiral iteration an error occurred
  - Who was the pre-last hop
  - Who was the next-hop when forwarding failed

```
+-----+  REQ a  +--------+ REQ branch0  +----+
| UAC |-------->| ...... |------------->|UAS1|
+-----+         |        |<--- 500 -----+----+
                |        |
                | proxy1 | REQ branch1              +--------+
                |        |------------------------->| proxy2 |--+
                |        | REQ 1.2.1 +----+         +--------+  |
             +->| ...... |---------->|UAS2|                     |
             |  +--------+           +----+                     v
             |                             REQ br1.2            |
             +---------------------<----------------------------+
```

Troubleshooting Proposal

- Operators do not know what is going wrong:
  - servers causing an error located on CP or belonging to a different administrative domain
  - users cannot report to operator on what is happening
- Proposal: take a lesson from email and include original message in replies – it includes all one needs to know
- Status: Already deployed at iptel.org, automated troubleshooting would take standardization

Concluding Observations

- Basic VoIP & complementary services up and running.
- Performance essential to survival of critical situations such as mis-configured networks and to avoidance of too many servers. Denial of Service still a pending challenge.
- Request-routing flexibility in servers essential to building services, but it takes troubleshooting facilities.
- Improvement place for phone implementations still exists: NAT traversal support, plug-and-play configuration, DNS fail-over.

Information Resources

- Email: jiri@iptel.org
- IP Telephony Information: http://www.iptel.org/info/
- SIP Services: http://www.iptel.org/user/
- SIP Express Router: http://www.iptel.org/ser/
- Related RFCs and Internet Drafts:

  http://www.iptel.org/info/
    - NATs: draft-ietf-sipping-nat-scenarios-00.txt
    - Diagnostic:draft-kuthan-sipping-diag-00.txt