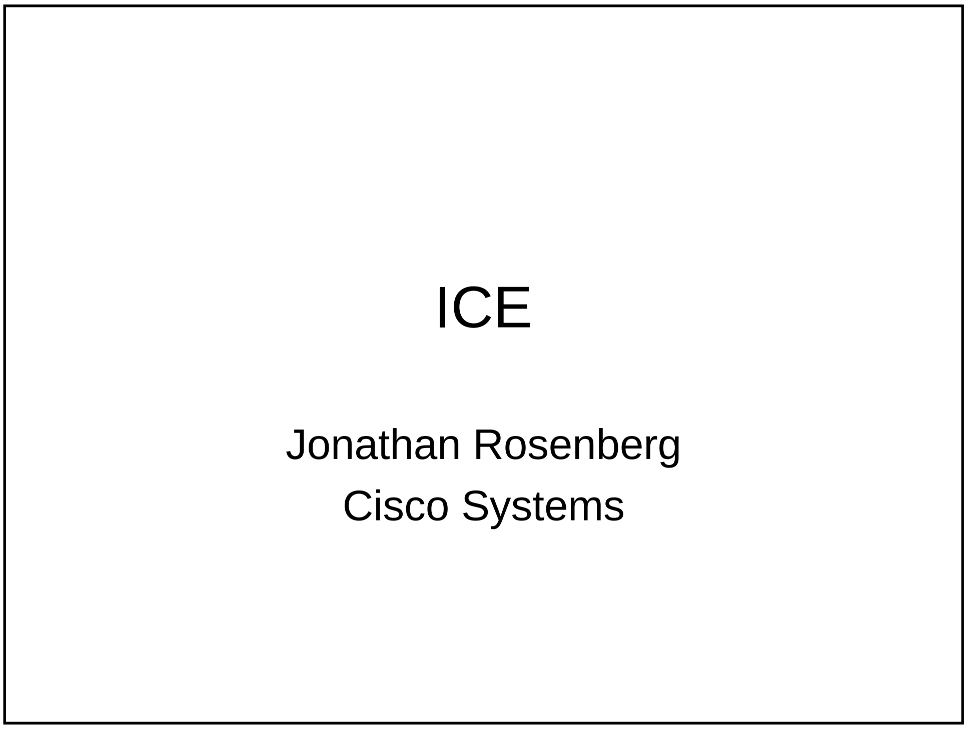
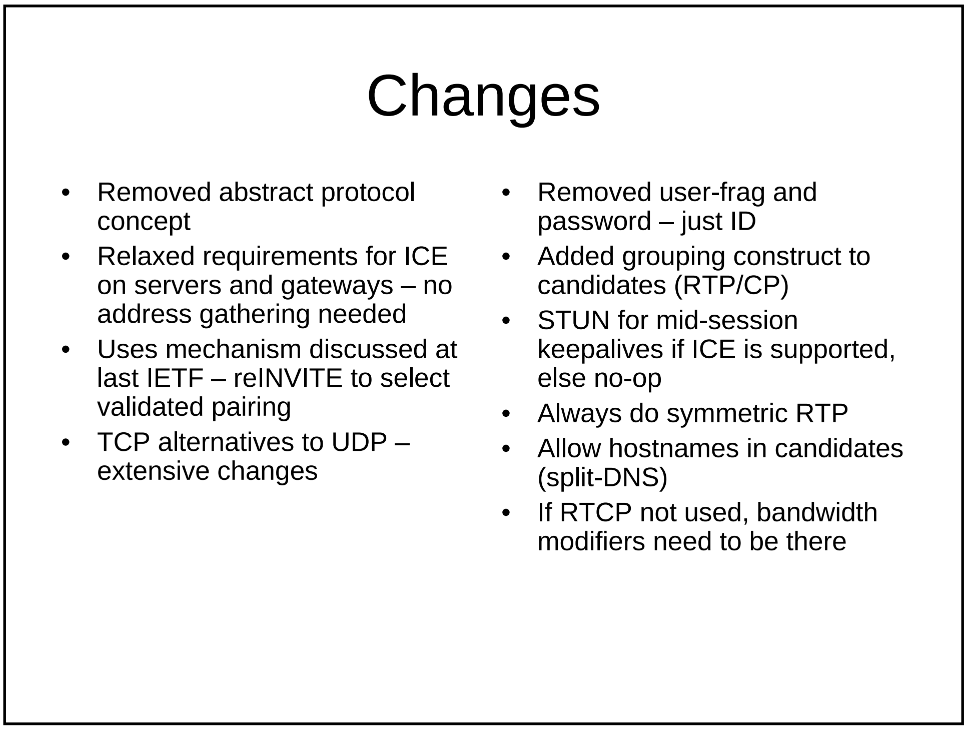


ICE

Jonathan Rosenberg
Cisco Systems



Changes

- Removed abstract protocol concept
- Relaxed requirements for ICE on servers and gateways – no address gathering needed
- Uses mechanism discussed at last IETF – reINVITE to select validated pairing
- TCP alternatives to UDP – extensive changes
- Removed user-frag and password – just ID
- Added grouping construct to candidates (RTP/CP)
- STUN for mid-session keepalives if ICE is supported, else no-op
- Always do symmetric RTP
- Allow hostnames in candidates (split-DNS)
- If RTCP not used, bandwidth modifiers need to be there

ICE Issue 1: STUN Floods

- Current algorithm does all connectivity checks in parallel
  - Number of checks = 2*interfaces*IP-versions*(STUN-servers + TURN servers)
  - Can be really big
- Consequences
  - Network bandwidth
  - NAT overload – reverting to symmetric behavior or refusing to create bindings
- Needs to be fixed

Proposed Fix

- Each side computes an absolute ordering of pairings
- STUN checks are rate limited like RTCP
- Each side does checks starting with highest priority, at maximum rate
- Once a check succeeds, stop and do an updated offer after Tb seconds
  - Eliminates un-needed checks
  - Tb deals with packet losses on higher priority checks – maybe 1 second or so
- Proposal: adopt?
  - What should rate limits be?

Issue 2: TCP or not TCP

- Lots of text added to deal with TCP
  - Significantly different than UDP – connections are not the same as pairings since you can’t do simultaneous open successfully in TCP
- RTP over TCP is of questionable value
- But, ICE really needs to make VoIP “just work” and thus should be aggressive with traversal
- Proposal:
  - Move it to separate document, progressed pretty much in parallel
  - Interoperability is easy

Issue 3: Default Timers

- Current timers
  - Tu: time to wait for active address to validate before an update (3s)
  - Tg: time to final updated offer (50s)
- Tg seems too large – set based on SIP default timers
- Intimately related to issue 1

Issue 4: STUN authentication and SIPS

- Current usage of STUN authentication and sips is vague
- Doing it is better than not
- But how likely are the attacks if its not there?
  - DoS attacks not possible as in regular STUN, even without crypto
  - Stealing media streams possible, but hard to coordinate (prevented with crypto)
- What happens if one side challenges but other side doesn’t respond? BAD
- Proposal:
  - Discuss threats (obviously)
  - STUN auth is MUST implement, SHOULD use
  - sips is SHOULD use

Issue 5: Normative Dependencies (again)

- Question from Francois around STUN – RFC 3489 or RFC 3489bis?
  - Timeframe on bis is questionable
  - Can specify behaviors using 3489 as basis
  - Propose: 3489