Abstract

STUN là một giao thức phục vụ như một công cụ cho các giao thức khác trong việc truyền tải thông tin NAT. Nó có thể được sử dụng bởi một endpoint để xác định địa chỉ IP và cổng được cấp phát bởi NAT (IP Public vs cổng ánh xạ với địa chỉ private). Nó cũng có thể được sử dụng để kiểm tra kết nối giữa hai thiết bị đầu cuối, và như là một giao thức duy trì để duy trì các ràng buộc NAT. STUN hoạt động với nhiều kiể NAT hiện có và không yêu cầu bất kỳ hành vi đặc biệt nào từ chúng.

STUN không phải là giải pháp truyền tải qua NAT. Thay vào đó, nó là một công cụ được sử dụng trong bối cảnh của một giải pháp traversal NAT. Đây là một thay đổi quan trọng so với phiên bản (RFC 3489), đã trình bày STUN như một giải pháp hoàn chỉnh.

1. Giới thiệu

Giao thức Session Traversal Utilities cho NAT(STUN), cung cấp một công cụ để xử lý với NAT. Nó cung cấp một phương tiện cho một điểm cuối để xác định địa chỉ IP và cổng được cấp phát bởi NAT tương ứng với địa chỉ IP và cổng riêng của nó. Nó cũng cung cấp một cách để giữ cho ánh xạ NAT đó được tồn tại. Với một số mở rộng, giao thức có thể được sử dụng để thực hiện kiểm tra kết nối giữa hai điểm cuối hoặc để chuyển tiếp các gói giữa hai điểm cuối

Để phù hợp với tính chất công cụ của nó, đặc tả này định nghĩa một định dạng gói mở rộng, xác định hoạt động trên một số giao thức truyền tải và cung cấp hai dạng xác thực.

STUN được dự định sẽ được sử dụng trong ngữ cảnh của một hoặc nhiều giải pháp truyền tải NAT. Các giải pháp này được gọi là cách sử dụng của STUN. Mỗi cách sử dụng mô tả cách STUN được sử dụng để đạt được giải pháp NAT traversal .Thông thường một cách sử dụng cho biết khi nào các thông điệp STUN được gửi đi, các thuộc tính tùy chọn để bao gồm, máy chủ nào được sử dụng và cơ chế xác thực nào sẽ được sử dụng. Thiết lập kết nối tương tác (ICE) [là một cách sử dụng STUN. SIP Outbound là một cách sử dụng STUN khác. Trong một số trường hợp, việc sử dụng sẽ yêu cầu tiện ích mở rộng cho STUN. Tiện ích mở rộng STUN có thể ở dạng phương thức, thuộc tính hoặc mã phản hồi lỗi mới. Bạn có thể tìm thêm thông tin về cách sử dụng STUN trong Phần 14.

1. Sự phát triển từ RFC 3489

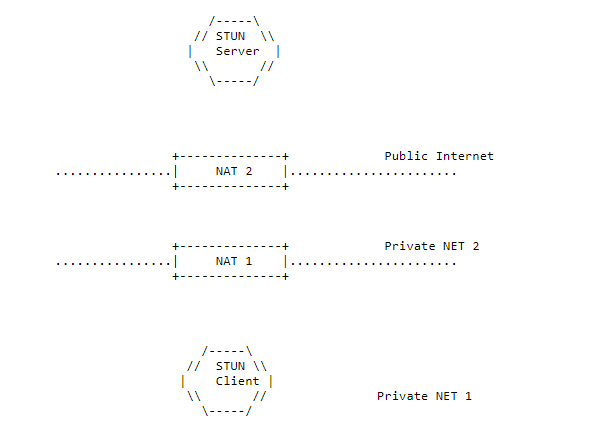
STUN ban đầu được định nghĩa trong RFC 3489 [RFC3489]. Đặc điểm kỹ thuật đó, đôi khi được gọi là "STUN cổ điển", đại diện cho chính nó như là một giải pháp hoàn chỉnh cho vấn đề NAT traversal. Trong giải pháp đó, một khách hàng sẽ phát hiện ra nó nằm sau NAT, xác định kiểu NAT của nó, khám phá địa chỉ IP và cổng của nó ở phía bên ngoài NAT ngoài cùng, sau đó sử dụng địa chỉ IP và cổng đó trong cơ thể các giao thức, làm Giao thức khởi tạo phiên (SIP) [RFC3261]. Tuy nhiên, kinh nghiệm kể từ khi công bố RFC 3489 đã phát hiện ra rằng STUN cổ điển chỉ đơn giản là không hoạt động đủ tốt để trở thành một giải pháp có thể triển khai. Địa chỉ và cổng học qua STUN cổ điển đôi khi có thể sử dụng để liên lạc với một người ngang hàng và đôi khi không. STUN cổ điển không cung cấp cách nào để khám phá xem liệu nó có thực sự hoạt động hay không, và nó không cung cấp biện pháp khắc phục nào trong trường hợp nó không hoạt động. Hơn nữa, thuật toán STUN cổ điển để phân loại các kiểu NAT đã được tìm thấy là bị lỗi, vì nhiều NAT không phù hợp với các loại được định nghĩa ở đó.

STUN cổ điển cũng có lỗ hổng bảo mật - kẻ tấn công có thể cung cấp cho khách hàng địa chỉ được ánh xạ không chính xác theo các cấu trúc liên kết và ràng buộc nhất định và điều này về cơ bản không thể giải quyết được bằng bất kỳ phương tiện mật mã nào. Mặc dù vấn đề này vẫn còn với đặc tả (STUN RFC5489) này, các cuộc tấn công này hiện nay được giảm thiểu thông qua việc sử dụng các giải pháp hoàn chỉnh hơn, sử dụng STUN.

Vì những lý do này, RFC 3489 đã lỗi thời, và thay vào đó mô tả STUN như một công cụ được sử dụng như một phần của giải pháp NAT traversal hoàn chỉnh. ICE là một giải pháp NAT traversal hoàn chỉnh cho các giao thức dựa trên phương pháp cung cấp / trả lời, chẳng hạn như SIP. SIP Outbound là một giải pháp hoàn chỉnh cho việc truyền tín hiệu SIP, và nó sử dụng STUN theo một cách rất khác. Mặc dù có thể một giao thức có thể tự sử dụng STUN (STUN cổ điển) làm giải pháp truyền tải, việc sử dụng như vậy không được mô tả ở đây và được khuyến khích mạnh mẽ vì các lý do được mô tả ở trên.

Giao thức được mô tả ở đây chỉ thay đổi một chút so với STUN cổ điển. Giao thức hiện chạy trên TCP ngoài UDP. Khả năng mở rộng đã được thêm vào giao thức theo cách có cấu trúc hơn. Một cơ chế sử dụng magic-cookie để phân tách STUN với các giao thức ứng dụng đã được thêm vào bằng cách lấy 32 bit từ transaction ID được định nghĩa trong RFC 3489, cho phép thay đổi tương thích ngược.Mapped address được mã hóa bằng cách sử dụng XOR. Có những thay đổi nhỏ khác. Xem Phần 19 để có danh sách đầy đủ hơn.

1. Tổng quan về hoạt động



Trong hình trên có 2 thực thể được gọi là STUN agent thực hiện giao thức STUN. Client được kết nối với mạng private 1. Mạng này kết nối với mạng private 2 qua NAT 1. Mạng private 2 kết nối với Internet thông qua NAT 2. Agent phía trên trong hình là máy chủ và nằm tại Internet .

STUN là giao thức máy client-server. Nó hỗ trợ hai loại bản tin trao đổi . Một là bản tin request/response trong đó một client gửi một request đến một máy chủ và máy chủ trả về một response. Thứ hai là bản tin báo hiệu trong đó agent - client hoặc server - gửi một dấu hiệu.Loại bản tin này không có phản hồi. Cả hai loại bản tin trao đổi bao gồm ID giao dịch, là số 96 bit được chọn ngẫu nhiên. Với các bản tin request/response , transaction ID cho phép client liên kết bản tin resquest với bản tin response. Với các bản tin báo hiệu thì transaction ID đóng vai trò trợ giúp gỡ lỗi.

Tài liệu này định nghĩa một phương thức gọi là Binding. Phương thức Binding có thể được sử dụng trong các bản tin resquest/response hay bản tin báo hiệu. Khi được sử dụng trong các bản tin resquest/response, phương thức Binding có thể được sử dụng để xác định ánh xạ địa chỉ và cổng của client sang địa chỉ public. Khi được sử dụng trong cả resquest/response hoặc trong các bản tin báo hiệu, phương thức Binding cũng có thể được sử dụng để giữ cho các "ánh xạ" này tồn tại.

Trong các bản tin Binding request / response, một Binding request được gửi từ một client đến một máy chủ STUN. Khi Binding request đến máy chủ STUN, nó có thể đã đi qua một hoặc nhiều NAT . Khi bản tin Binding request đi qua NAT, NAT sẽ sửa đổi **transport address** (có nghĩa là địa chỉ IP nguồn và cổng nguồn) của gói. Do đó, IP nguồn và port nguồn mà server nhận được sẽ là địa chỉ **IP public và port được NAT gần nhất với server tạo ra**. Máy chủ STUN sao chép địa chỉ IP nguồn và port nguồn vào thuộc tính XOR-MAPPED- ADDRESS trong bản tin Binding response và gửi bản tin này cho client. Bằng cách này, client có thể biết địa chỉ public được cấp phát bởi NAT ngoài cùng đối với STUN server.

Trong một số trường hợp, STUN được ghép với các giao thức khác. Trong trường hợp này, phải có cách kiểm tra gói tin và xác định xem đó có phải là gói STUN hay không. STUN cung cấp ba trường trong STUN header với các giá trị cố định có thể được sử dụng cho mục đích này. Nếu header là không đủ, thì các gói STUN cũng có thể chứa giá trị FINGERPRINT, có thể được sử dụng để phân biệt .

STUN định nghĩa một tập các thủ tục tùy chọn mà một cách sử dụng có thể quyết định sử dụng, được gọi là các cơ chế. Các cơ chế này bao gồm phát hiện DNS, kỹ thuật chuyển hướng đến một máy chủ thay thế, một thuộc tính FINGERPRINT để phân kênh và hai trao đổi xác thực và toàn vẹn thông điệp. Cơ chế xác thực xoay quanh việc sử dụng tên người dùng, mật khẩu và giá trị message-integrity. Hai cơ chế xác thực, cơ chế long-term credential và short-term credential, được định nghĩa tài liệu này.

Trong cơ chế long-term credential, client và server chia sẻ tên người dùng và mật khẩu được cấp trước. Trong cơ chế short-term credential, client và server trao đổi tên người dùng và mật khẩu thông qua một số phương thức ngoài trước khi thực hiện các bản tin của giao thức STUN.

1. Các định nghĩa

STUN Agent: Một STUN agent là một thực thể sử dụng giao thức STUN. Thực thể đó có thể là client hoặc server.

STUN Client: A STUN client là một thực thể gửi các bản tin request và nhận về các bản tin trả lời. Nó cũng có thể gửi một bản tin báo hiệu.

STUN Server: A STUN server is an entity that receives STUN requests and sends STUN responses. A STUN server can also send indications.

Transport Address: là sự kết hợp của IP address và port.

Reflexive Transport Address: là một Transport Address được tạo ra bởi NAT gần server nhất.

Long-Term Credential: Username va password được liên kết đại diện cho bí mật được chia sẻ giữa client và server. Long-term credentials thường được cấp cho khách hàng khi người đăng ký dịch vụ và vẫn tồn tại cho đến khi người đăng ký rời khỏi dịch vụ hoặc thay đổi thông tin xác thực.

Short-Term Credential: Tên người dùng tạm thời và mật khẩu được liên kết đại diện cho bí mật được chia sẻ giữa client và server. Short-Term Credential thu được thông qua một số loại giao thức giữa client và server, trước khi trao đổi qua giao thức STUN. Thông tin xác thực ngắn hạn có phạm vi thời gian rõ ràng, có thể dựa trên một lượng thời gian cụ thể (chẳng hạn như 5 phút) hoặc trên một sự kiện. Phạm vi cụ thể của Short-Term Credential được xác định bằng cách sử dụng ứng dụng.

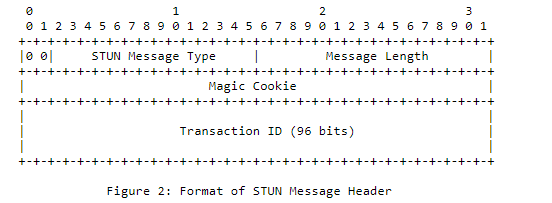
STUN Indication: là một loại bản tin báo hiệu mà không nhận lại response.

Attribute: Các thuộc tính được chia thành hai loại: bắt buộc và tùy chọn. STUN agents có thể bỏ qua các thuộc tính tùy chọn, nhưng không thể xử lý thành công một message nếu nó không chứa các thuộc tính bắt buộc.

RTO: Retransmission TimeOut, xác định khoảng thời gian ban đầu giữa việc truyền yêu cầu và lần truyền lại đầu tiên của yêu cầu đó.(3s).

1. Cấu trúc bản tin STUN

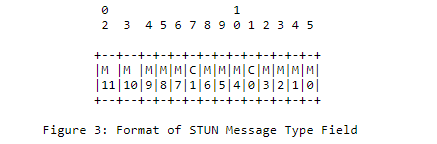
Các thông điệp STUN được mã hóa dưới dạng nhị phân sử dụng định dạng.



2 bit quan trọng nhất của mỗi bản tin của giao thức STUN PHẢI là 0. Điều này có thể được sử dụng để phân biệt các bản tin STUN với các giao thức khác khi STUN được ghép kênh với các giao thức khác trên cùng một cổng.

Message type bao gồm (request, success response, failure response, or indication). Có 4 message type nhưng chỉ có 2 loại giao dịch trong các giao dịch của giao thức STUN đó là request/response transactions và indication transactions.

Cấu trúc của STUN Message Type:



Các bit M là các bit cố định. C1C0 là đại diện cho 4 loại Message

0x00: request

0x01: indication

0x10: success reponse

0x11: failure reponse

Trường magic cookie PHẢI chứa giá trị cố định 0x2112A442. Trong RFC 3489, trường này là một phần của transaction ID. **Nó hỗ trợ phân biệt các gói STUN từ các gói của các giao thức khác khi STUN được ghép kênh với các giao thức khác trên cùng một cổng.**

Transaction ID là 1 mã đinh danh 96 bit. Nó được sử dụng để nhận dạng duy nhất các giao dịch của STUN. Các giao dịch request/response thì transaction ID được lựa chọn bởi client và server sử dụng ID này để trả lời. Với bản tin indications, transaction ID được chọn bởi bên gửi.

1. Các thủ tục cơ bản của giao thức
   1. Hình thành 1 bản tin request hoặc indications

Khi tạo một bản tin request hoặc indications thì nó phải tuân thủ cấu trúc header ở phần 5. Nếu tác nhân gửi request, NÊN thêm thuộc tính SOFTWARE vào request. Agents có thể bao gồm thuộc tính SOFTWARE trong bản tin indications, tùy thuộc vào phương thức. Với bản tin Binding request không có xác thực thì không yêu cầu thêm thuộc tính.

Tất cả bản tin STUN được gửi qua UDP NÊN nhỏ hơn MTU của đường dẫn. Nếu MTU của đường dẫn không xác định, bản tin STUN NÊN nhỏ hơn 576 byte cho IPv4 và 1280 byte cho IPv6.

* 1. Gửi bản tin request hoặc indications

Tài liệu này chỉ định cách gửi các thông điệp STUN qua UDP, TCP hoặc TLS-over-TCP; các giao thức vận tải khác có thể được thêm vào trong tương lai. Việc sử dụng STUN phải chỉ định giao thức truyền tải nào được sử dụng và cách agent xác định địa chỉ IP và cổng của người nhận.

* + 1. Gửi thông qua UDP

Khi client sử dụng UDP để truyền tải thông tin STUN, thì chúng có thể bị loại bỏ bởi mạng(firewall). Độ tin cậy của các giao dịch request / response được thực hiện thông qua việc truyền lại bản tin request của chính client. Indications không được truyền lại; do đó, các giao dịch trên UDP không đáng tin cậy.

Client NÊN gửi lại một bản tin request sau một khoảng thời gian RTO ("Retransmission TimeOut"), tăng gấp đôi sau mỗi lần truyền lại. RTO là một ước tính của Round Trip Time (RTT). Giá trị ban đầu NÊN được cấu hình lớn hơn 500ms. Giá trị của RTO KHÔNG NÊN được làm tròn đến giây gần nhất. Thay vào đó, độ chính xác 1 ms NÊN được duy trì.

Giá trị cho RTO NÊN được lưu trữ bởi client sau khi hoàn thành giao dịch và được sử dụng làm giá trị bắt đầu cho RTO cho giao dịch tiếp theo cho cùng một máy chủ .Giá trị RTO NÊN bị loại bỏ sau 10 phút.

Truyền lại tiếp tục cho đến khi nhận được phản hồi hoặc cho đến khi tổng số yêu cầu Rc đã được gửi. Rc NÊN được đặt là 7. Nếu sau yêu cầu cuối cùng, thời gian bằng Rm lần RTO đã trôi qua mà không có phản hồi client NÊN coi giao dịch đã thất bại. Rm được đặt là 16.

* + 1. Gửi qua TCP hoặc TLS qua TCP

Client mở một kết nối TCP đến server.

Trong một số cách sử dụng STUN, STUN được gửi dưới dạng giao thức duy nhất qua kết nối TCP. Máy khách CÓ THỂ gửi nhiều giao dịch qua một kết nối TCP (hoặc TLS-over-TCP) duy nhất, và nó CÓ THỂ gửi một request khác trước khi nhận được phản hồi trước đó. Client NÊN giữ kết nối mở cho đến khi nó:

* không có thêm request hoặc chỉ dẫn nào để gửi qua kết nối đó
* không có kế hoạch sử dụng bất kỳ tài nguyên nào đã được học qua các request được gửi qua kết nối đó

Server NÊN giữ kết nối đang mở và để máy khách đóng nó, trừ khi máy chủ đã xác định rằng kết nối đã hết time out.

* 1. Nhận bản tin STUN
     1. Xử lý 1 request

Nếu request chứa một hoặc nhiều thuộc tính KHÔNG BIẾT, server trả lời bằng bản tin error response với mã lỗi 420 (Thuộc tính không xác định) và bao gồm thuộc tính UNKNOWN-ATTRIBUTES trong response liệt kê các thuộc tính bắt buộc chưa biết.

Khi chạy trên UDP, một request nhận được bởi server có thể là request đầu tiên của một giao dịch hoặc một lần truyền lại.