AP1Q5 - Skype

Vinicius Gasparini

REC - BCC - UDESC - 2020

1 História e histórico

Fundada em 2003 [1], a Skype é uma empresa desenvolvedora de um *software* homônimo que realizava chamadas de voz via protocolo P2P. Em 2009, a Skype então já estabelecida no mercado com quase 39 milhões de usuários [2] é comprada pelo eBay [3]. Por fim em 2011 a Microsoft adquire a companhia por US$ 8.56 bilhões [4].

Desenvolvido pela mesma equipe do Kazaa [5], a Skype modelou um protocolo voIP (*voice over IP*) proprietário chamado *Skype Protocol* que os posicionou como o precursores dentro do segmento serviço de telefonia via *peer-to-peer* IP [6]. Essa arquitetura operava com base em três princípios:

- Supernodos: um supernodo dentro da arquitetura poderia ser qualquer ponta da rede que possuísse boa largura de banda, não possuir restrição de **firewall** e suficiente poder computacional para lidar com diversos processos de conexão.
- Nodos de usuário: cada cliente. Essas instâncias da rede permitiam os usuários se conectarem por meio de supernodos.
- Servidor de *login*: informações de usuários que eram descentralizadas e distribuídas pela rede.

Em 2012 a Microsoft alterou essa estrutura para que os supernodos fossem hospedados em seus servidores para garantir estabilidade e escalar a disponibilidade das redes para chamadas, visto que foi nessa época implementado o serviço de video chamada.

A última atualização e a qual o presente trabalho tem foco é a realizada em 2014 ao migrar a operação para o novo *Microsoft Notification Protocol 24*.

O MSNP é um protocolo proprietário da Microsoft que vem sido desde 1999 e implementado nos produtos que utilizam troca de mensagens da empresa. A Tabela 1 compara as diferentes versões do protocolo e destaca as principais alterações.

Versão	Ano	Diferencial
MSNP1	1999	Criado para uso no MSN 1
MSNP8	2003	Integrado ao protocolo autenticação e suporte a transferência de vídeo e aúdio
MSNP9	2003	Suporte a streaming de imagem (WMV) e arquivos via NAT melhorado
MSNP13	2005	Funcionalidade de sincronia da lista de contatos via requisição SOAP
MSNP15	2006	Modificado mecanismo de autenticação e persistência dos dados de perfil
MSNP16	2007	Múltiplos logins simultâneos
MSNP24	2014	Adoção do protocolo pelo Skype

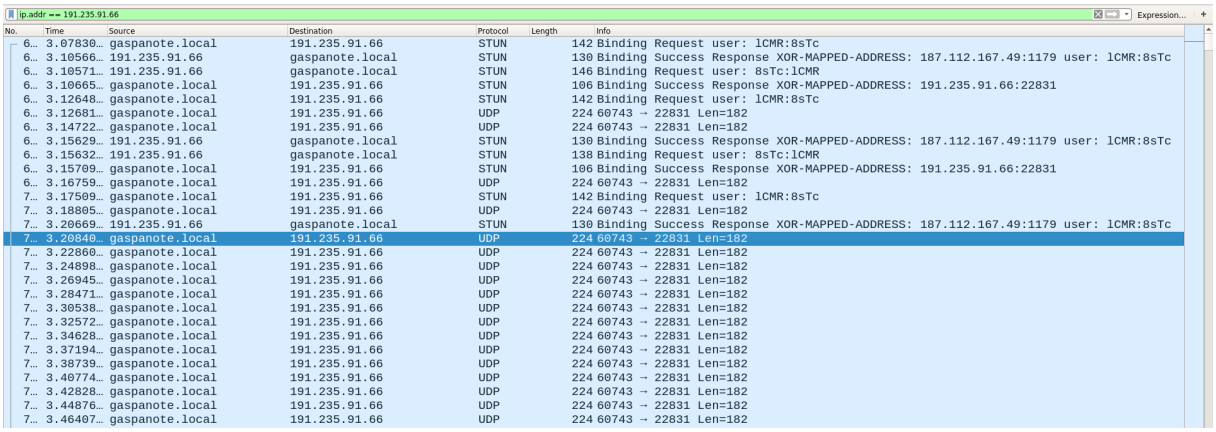
Tabela 1: Comparação entre versões do MSNP

2 Definições

A documentação para tal protocolo é bem escassa por se tratar de um protocolo closed source, mas conforme podemos observar neste comentário de 2018 de um moderador Microsoft [7], com base no livro *Practical Packet Analysis* [8] e na captura realizada pela comunidade e disponibilizada nos documentos do Wireshark [9], o protocolo **MSNP** utiliza tanto **TCP** quanto **UDP** para na camada de transporte. A nível de aplicação, visto que a nova interface cliente também está disponível via *web browser application*, utiliza-se **HTTP** e **HTTPS**.

Durante testes com Wireshark, encontrei também o uso do protocolo **STUN** (*Session Traversal Utilities for NAT*). O STUN é conforme RFC3489, "um protocolo leve que permite aplicações UDP a realizarem descoberta e os tipos de NATs e firewalls entre as mesmas e a Internet".

Quanto ao diagrama de sequência, pouco se tem informação de quem são os atores envolvidos nas transações. O que se pode observar na captura da Figura 1 é que algumas chamadas STUN são realizadas a fim de iniciar a conexão utilizando o protocolo MSNP. Este por sua vez é descrito no documento produzido pela comunidade utilizando engenharia reversa no Windows Live Mail [10] (Figura 2). O que podemos interpretar portanto acerca das interações é como segue na Figura 3.



No.	Time	Source	Destination	Protocol	Length	Info
6…	3.07830…	gaspanote.local	191.235.91.66	STUN	142	Binding Request user: lCMR:8sTc
6…	3.10566…	191.235.91.66	gaspanote.local	STUN	130	Binding Success Response XOR-MAPPED-ADDRESS: 187.112.167.49:1179 user: lCMR:8sTc
6…	3.10571…	191.235.91.66	gaspanote.local	STUN	146	Binding Request user: 8sTc:lCMR
6…	3.10665…	gaspanote.local	191.235.91.66	STUN	106	Binding Success Response XOR-MAPPED-ADDRESS: 191.235.91.66:22831
6…	3.12648…	gaspanote.local	191.235.91.66	STUN	142	Binding Request user: lCMR:8sTc
6…	3.12681…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
6…	3.14722…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
6…	3.15629…	191.235.91.66	gaspanote.local	STUN	130	Binding Success Response XOR-MAPPED-ADDRESS: 187.112.167.49:1179 user: lCMR:8sTc
6…	3.15632…	191.235.91.66	gaspanote.local	STUN	138	Binding Request user: 8sTc:lCMR
6…	3.15709…	gaspanote.local	191.235.91.66	STUN	106	Binding Success Response XOR-MAPPED-ADDRESS: 191.235.91.66:22831
6…	3.16759…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
7…	3.17509…	gaspanote.local	191.235.91.66	STUN	142	Binding Request user: lCMR:8sTc
7…	3.18805…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
7…	3.20669…	191.235.91.66	gaspanote.local	STUN	130	Binding Success Response XOR-MAPPED-ADDRESS: 187.112.167.49:1179 user: lCMR:8sTc
7…	3.20840…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
7…	3.22860…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
7…	3.24898…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
7…	3.26945…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
7…	3.28471…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
7…	3.30538…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
7…	3.32572…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
7…	3.34628…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
7…	3.37194…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
7…	3.38739…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
7…	3.40774…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
7…	3.42828…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
7…	3.44876…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182
7…	3.46407…	gaspanote.local	191.235.91.66	UDP	224	60743 → 22831 Len=182

Figura 1: Interface do Wireshark após captura de uma chama via Skype Web

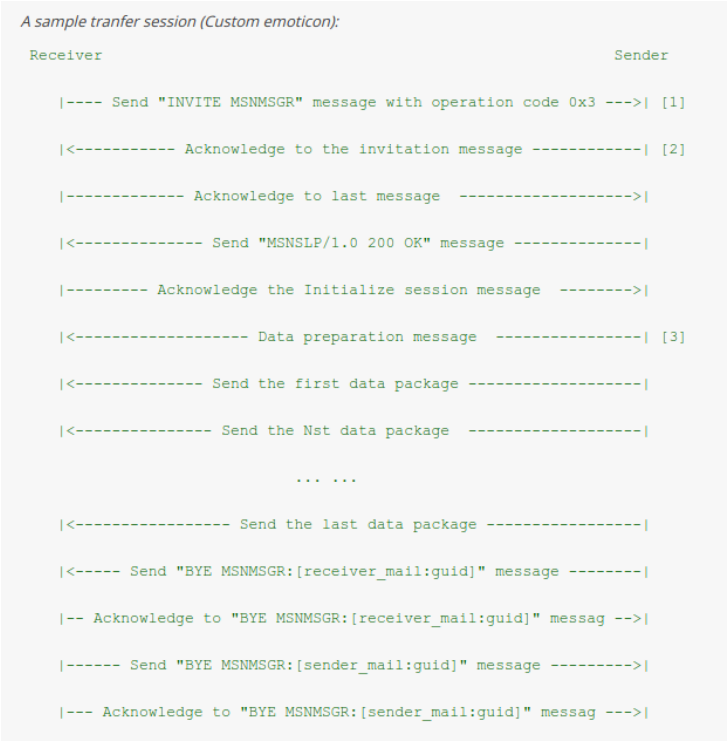


Figura 2: Diagrama sequência do MSNP21

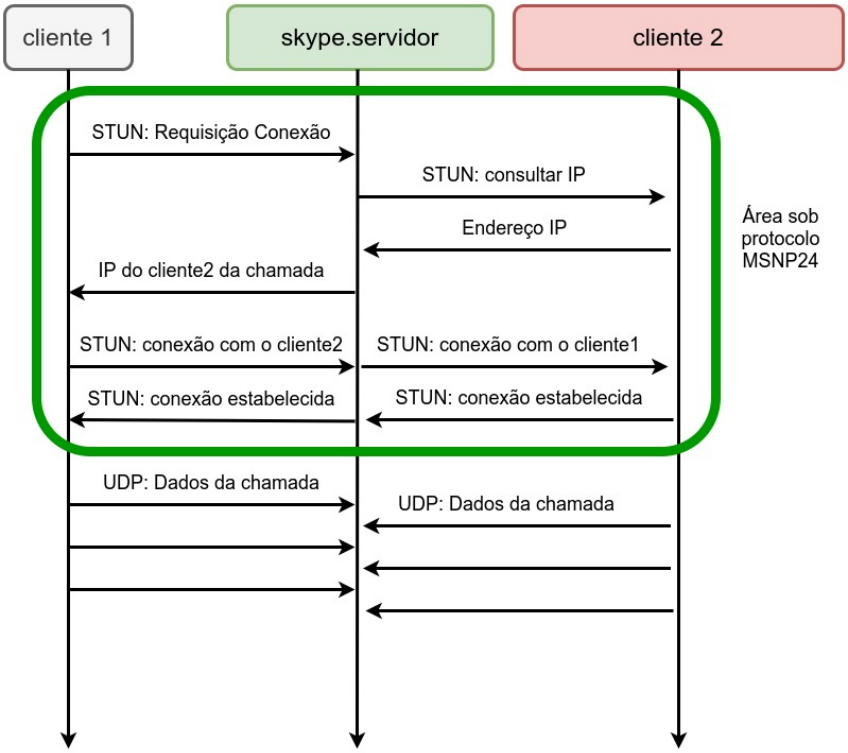


Figura 3: Diagrama sequência de uma chamada.

Referências

[1] "Skype: About." `https://www.skype.com/en/about/`. Acessado: 21-12-2020.

[2] "Gigaom: Skype by the numbers." `https://gigaom.com/2010/04/20/skype-q4-2009-number/`. Acessado: 21-12-2020.

[3] "Sfgate: Ebay to buy skype for $2.6 billion." `https://www.sfgate.com/technology/article/EBay-to-buy-Skype-for-2-6-billion-2570068.php`. Acessado: 21-12-2020.

[4] "Wired: Microsoft buys skype for $8.5 billion.." `https://www.wired.com/2011/05/microsoft-buys-skype-2/`. Acessado: 21-12-2020.

[5] "Arstechinica: The strange story of skype." `https://arstechnica.com/information-technology/2018/09/skypes-secrets/2/`. Acessado: 21-12-2020.

[6] S. Baset and H. Schulzrinne, "An analysis of the skype peer-to-peer internet telephony protocol," in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies, 23-29 April 2006, Barcelona, Catalunya, Spain*, IEEE, 2006.

[7] "Tcp vs udp in skype for business online." `https://answers.microsoft.com/en-us/msoffice/forum/all/tcp-vs-udp-in-skype-for-business-online/f5de594a-06ff-4e9f-9063-1729480e2187`. Acessado: 21-12-2020.

[8] C. Sanders, *Practical Packet Analysis: Using Wireshark to Solve Real-world Network Problems*. No Starch Press Series, No Starch Press, 2007.

[9] "Wireshark: Msn messenger service (msnms)." `https://wiki.wireshark.org/MSNMS`. Acessado: 21-12-2020.

[10] "Research on binary header that wlm 2009 used for its p2p protocol." `https://code.google.com/archive/p/msnp-sharp/wikis/KB_MSNC12_BinaryHeader.wiki`. Acessado: 21-12-2020.