Reversing the Original Xbox Live Protocols

DEF CON 30

monocasa

This Talk

- Background
- The Protocols Themselves
- Approach to Reversing
- Replacement Server Architecture
- Ruminations on Supporting Abandoned Network Software
- Demo & Questions

Background

- Who am I?
- monocasa
- An engineer with experience in binary reversing, firmware development, cloud based device and identity management, and custom tunneling of IP
- monocasa@xombie.org

Background (cont.)

- Xbox Live
- This talk is on the original service for the original console – Nov 5, 2002 until Apr 15 2010

Background (cont.)

- First modern video game console network
- Cohesive service across games
- Expected broadband
- Enough local storage to update games properly and download addons
- Moderated

The Protocols

- IPv4
- DNS
- Kerberos
- UDP Port 3074 – Custom VPN Protocol

Crypto Primitives

- MD5
- SHA1
- RC4
- DES
- 3DES

DNS

- MACS.XBOXLIVE.COM
- AS.XBOXLIVE.COM
- TGS.XBOXLIVE.COM
- SG%d.XBOXLIVE.COM

DNS (contd.)

- Partner Net
- *.PART.XBOXLIVE.COM

Kerberos

- A lot like the 1980s version of Oauth
- ASN.1 (DER) instead of JSON/JWT
- “Tickets” instead of “Tokens”
- Doesn’t require public/private crypto, but instead based entirely around pre shared keys
- Heavily extensible with “pre auth” (pa) data fields

MACS

- Machine Account Creation Service
- Converts secrets allocated to console at manufacturing time into a Xbox Live user
- Protocol at this point ignores tickets, just using Kerberos AS-REQ/AS-REP to verify shared secret, and to shuffle data in undocumented pa fields

MACS (contd.)

- AS-REQ from console to server
- 206 – Client Version signed with secrets
- 204 – Pre Pre Auth – Hashed console serial and secrets
- 131 – Constraint on user data in a way that is meaningless
- 2 – Current timestamp encrypted and signed
- Otherwise standard AS-REQ

MACS (cont.)

- AS-REP server to console
- A ticket that gets thrown away
- PA 203 – Account Creation – contains machine user identity
- The box stores this info in unpartioned space on the local HDD

Users

- Xuid – 64 bit user id
- Gamertag – 14 byte ascii string
- Key – 16 byte shared secret
- Kerberos domain and realm
- Non machine users have flags and passcode

AS.XBOXLIVE.COM

- AS-REQ/AS-REP Pair
- Creates a “ticket granting ticket” when presented with cryptographic proof of knowledge of preshared key
- Similar to OAuth bearer token

AS (cont.)

- Can be a multi step process
- First creates TGT with machine user identity, then subsequent interactions present ticket and additional identities to be combined into the ticket

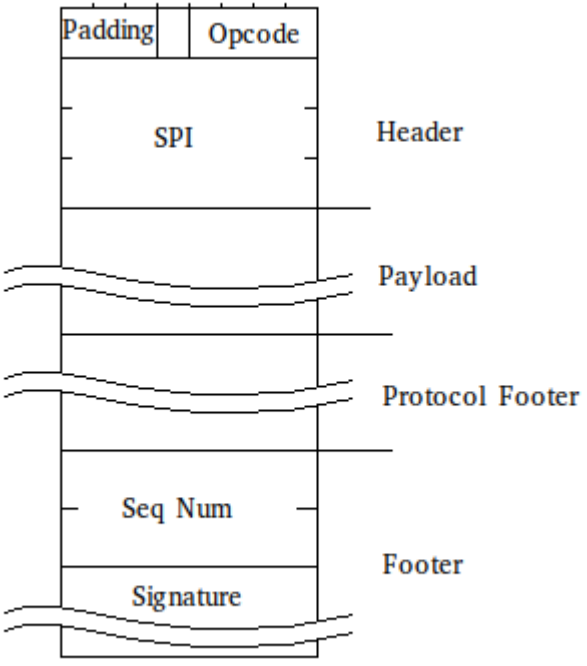
TGS

- Creates “service ticket” from “ticket granting ticket”
- PA 201 – Service Request
- PA 202 – Service Address

Secure Gateway

- Port 3074
- Custom VPN related to IPSEC
- Signs and encrypts almost all traffic

Secure Gateway Packet Format



Control Packets (Opcode 0)

- Initialization – Diffie Hellman, and paramter negotiation
- Shutdown
- Keep alives
- Event queue netogtiated within keep alives
- Throttling QoS

Other Opcodes

- UDP and TCP (but with cruft removed)
- Each are three opcodes, compression of port numbers
- VDP – Voice Data Protocol – UDP with unencrypted (but still signed) section for ‘lawful intercept’

Backend Services

- ~20 builtin services, most games only use a handful
- Presence service is always used
- Games can have custom services
- Halo 2 uses this custom service feature : (

Backend Services (cont.)

- Tend to be HTTP with custom binary payloads
- Strong signals that they’re primarily written against C# ASP.net servers (but probably would have been a beta version originally)

Backend Services (cont.)

- Presence: Always exists, client pushed events for starting a game, etc.
- Matchmaking: Creating querying for sessions
- Strings: internationalization and sanitization for moderation
- Feedback: moderation requests

Backend Services (cont.)

- Stats: Leaderboards
- Messaging: Messages from the system or other users
- Auto Update: Making sure a game is the same version for all players
- Teams: Joining a team

Joining a Game Session

- Use the matchmaking service to query for or create a session
- Session information contains host
- Host can be a dedicated server, or another client xbox
- Use more or less same VPN protocol to login to the host
- Don’t have to diffie hellman since the backend will just give you a key set to use as part of the session
- Uses same 3074 port as primary NAT punching technique

System Link

- Essentially the same protocols and flow
- Local broadcast packets encrypted with per game key to announce availbility as host
- Then clients will VPN into the host and game netowroking proceeds as normal from the game’s perspective
- Games in fact have no TCP stack other than on top of VPN protocol

Approach to Reversing

- A lot of staring at Ghidra
- Fun facts:
  ”Gauntlet Dark Legacy” ships with PDB on disk
  ”The Red Star” ships with symbol map file
  ”Star Wars Jedi Knight: Jedi Academy” is GPLed
  XONLINE and XNET sections are broken out
  All crypto is cleanly exported by the kernel

Replacement Server Architecture

- Xombie.org for alpha
- github.com/XombieOnline/xombie for AGPL source
- Massive simplification of what MS has to deal with, can probably live on a single box indefinitely
- Might split out pieces to allow blue/green deployments
- PRs are very welcome

Ruminations on Supporting Abandoned Networked Software

- Theory of Software Security Harm Reduction
- How Network Security Changes When All Users are Expected to Have Hacked Their Systems

Demo & Questions