

Decoding Brickstorms Garble strings

 medium.com/walmartglobaltech/decoding-brickstorms-garble-strings-b0a60828b3cc

Jason Reaves

December 4, 2025

We're powering the next great retail disruption. Learn more about us—

<https://www.linkedin.com/company/walmartglobaltech/>

From the recent attention to Brickstorm in public reporting, I analyzed the Brickstorm sample listed below that was recently uploaded to Virustotal:

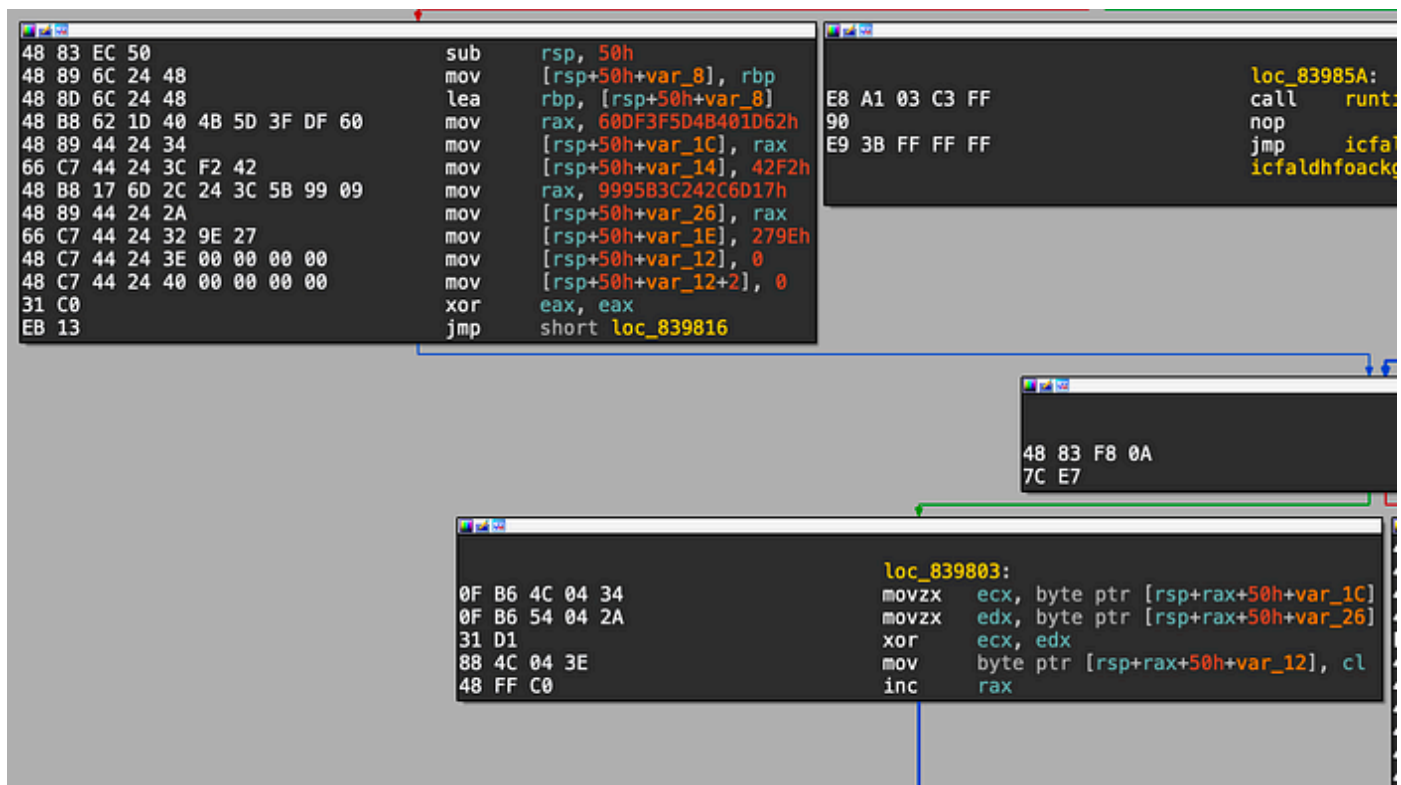
90b760ed1d0dcb3ef0f2b6d6195c9d852bcb65eca293578982a8c4b64f51b035

This sample was obfuscated via an open-source Golang tool called Garble[4] which was mentioned by Mandiant[1]. Mandiant also released a tool for decoding Garble strings[2] which was based on prior work by OALabs[3].

A quick note before diving in: I developed my string decoder independently, without first reviewing the work published by Google Threat Intelligence (GTI). Interestingly, my approach ended up aligning closely with their research on similar samples. I want to spotlight the method detailed in GTI's blog, which takes a deep dive into Garble's AST transformation code, a technique that offered valuable insight into how the obfuscation operates.

Join Medium for free to get updates from this writer.

The strings are mostly stack based although some of the larger ones reside elsewhere and are loaded.



For harvesting the bytecode sequences I decided to leverage a few possible yara rules which started strict and was relaxed as I continued my work:

```
= yara.compile(source=) = yara.compile(source=)
```

After retrieving all the matches and removing possible substrings I can emulate the code:

```
blobs = uniq_blobsout = : STACK= code_base = blob blobs: mu =
Uc(UC_ARCH_X86,UC_MODE_64) mu.mem_map(code_base, ) mu.mem_map(STACK, *)
blob = blob.matched_data+ mu.mem_write(code_base,
*) mu.mem_write(STACK, *(*)) mu.mem_write(code_base,blob)
mu.reg_write(UC_X86_REG_RSP,STACK+) mu.reg_write(UC_X86_REG_RBP,STACK+)
mu.reg_write(UC_X86_REG_ESP,STACK+) mu.reg_write(UC_X86_REG_EBP,STACK+)
mu.reg_write(UC_X86_REG_RAX,) : mu.emu_start(code_base, code_base+
(blob), timeout=) : a = mu.mem_read(STACK,*) a =
.join(a.split()) l = mu.reg_read(UC_X86_REG_RAX)+ (a) > : ((a)[-
1:]) out += a
```

I noticed afterwards that OALabs also used a hook but they did so to find the next call instruction:

```
def (uc, address, size, user_data): insn = (cs.(uc.(address, size), address))
#(f) if insn.mnemonic == : #() uc.()
```

I like this approach and will definitely leverage it later, I primarily used hooks for debugging my code in comparison:

```
() : ( %(address, size)) () : (user_data) (uc.reg_read(UC_X86_REG_RCX)) ( %
(address, size))
```

This won't get every single string as some are passed as offsets to the data residing in rodata section for longer pieces. Decoding:

Handshakeinvalid perspective-RTT ProtectedQUIC_GO_LOG_LEVELt fit into bitscannot encode
 %d in %d byteschacha2: wrong key sizechacha2: invalid buffer overlapchacha2: wrong
 HChaCha2 key sizeGODEBUG sys/cpu: value specified GODEBUG sys/cpu: unknown cpu feature
 GODEBUG sys/cpu: can disable avx512vaesavx512bitalgpcmulqdcchacha20poly1305: bad nonce
 passed to Sealchacha20poly1305: bad nonce passed to Openchacha20poly1305: invalid buffer
 overlapZ070cryptobyte: Builder is exceeding its fixed-size bufferbad input point: low
 order point notifybad record MACrecord overflowhandshake failureunsupported
 certificateexpired certificateillegal parameteraccess deniederror decrypting
 messageprotocol version supportedinappropriate fallbackunsupported extensionunrecognized
 namecertificate requiredGTLS , server
 CertificateVerifySERVER_HANDSHAKE_TRAFFIC_SECRETSERVER_TRAFFIC_SECRET_tls: invalid
 ServerKeyExchange messagec hs trafficc ap traffic mastertraffic updkey expansionserver
 finishedexpected an ECDSA public key, got %Texpected an RSA public key, got %Tunsupported
 signature algorithm: %vtls: unsupported public key: %Ttls: certificate private key (%T)
 does implement crypto.Signertls: certificate RSA key size too small supported signature
 algorithmst support the certificate custom signature algorithmstls: unable to generate
 random session ticket key: %vtls: received unexpected handshake message of type %T
 waiting %Tunknown cipher typeunknown cipher typetls: too many non-advancing recordstls:
 internal error: handshake should have had a resulttls: NextProtos too largetls: short
 from Rand: resumptiontls: received unexpected CertificateStatus messages identity changed
 during renegotiationtls: server advertised unrequested ALPN extensiontls: server resumed a
 session with a different cipher suites Finished message was incorrecttls: server selected
 unsupported compression tls: server selected unsupported grouptls: server sent a cookie in
 a normal ServerHellotls: server did a key sharetls: server selected an invalid PSKtls:
 invalid server key shareALPN negotiation failed. Server offered: %qtls: certificate used
 with invalid signature algorithmtls: invalid server finished hashtls: received a session
 ticket with invalid lifetimetls: internal error: failed to update binderstls: client
 offered old TLS version %#xtls: client does support uncompressed connectionstls: cipher
 suite supported by both client servertls: client certificate used with invalid signature
 algorithmtls: client\tls: failed to parse client certificate: tls: ECDHE curve supported
 by both client servertls: client sent unexpected early datatls: internal error: failed to
 clone hashc e traffictls: client indicated early data in second ClientHellotls: client
 offered -RTT data in second ClientHellotls: failed to sign handshake: tls: client
 certificate used with invalid signature algorithmtls: invalid client finished hashtls:
 supported elliptic curves offeredtls: failed to sign ECDHE parameters: tls: certificate
 used with invalid signature algorithmtls: missing ServerKeyExchange messagetls: internal
 error: unsupported curveserver finishedkey expansioncrypto/tls: ExportKeyingMaterial
 context too longtls: failed to create cipher encrypting ticket: qtls.ClientSessionState
 doesnt matchCRYPTO_BUFFER_EXCEEDEDCRYPTO_ERROR (%#x) (frame type: %#x) compatible QUIC
 version found (we support %s, server offered %s)t%s &wire.MaxDataFrame{MaximumData: %d}t%s
 &wire.DataBlockedFrame{MaximumData: %d}t%s &wire.MaxStreamsFrame{Type: uni, MaxStreamNum:
 %d}t%s &wire.StreamsBlockedFrame{Type: bidi, MaxStreams: %d}%d exceeds the maximum stream
 countstream data overflows maximum offset%d exceeds the maximum stream countwrong
 disable_active_migration: %d (expected empty)client sent an
 original_destination_connection_idexpected preferred_address to be %d long, %d
 bytesinitial_max_streams_bidi too large: %d (maximum %d)invalid value ack_delay_exponent:
 %d (maximum %d)TransportParameter BUG: transport parameter %d
 foundRetrySourceConnectionID: %s, Version Negotiation packet has a version list with an
 invalid such processinput/output error error child processesbad addressdevice resource
 busy such deviceis a directorytoo many filesfile too largeillegal too many linksnumerical
 argument out of domainfile name too longfunction implementeddirectory emptyidentifier
 removedlevel haltedexchange fullbad font file timer expiredmachine is on the network has

been severedprotocol errormultihop attemptedbad messagefile descriptor in bad accessing a
 corrupted shared libraryinvalid incomplete multibyte wide charactertoo many
 usersdestination address requiredprotocol availableprotocol supportedoperation
 supportedprotocol family supportedaddress family supported by protocoladdress already in
 ENETUNREACHENETRESETCONNABORTEDtransport endpoint is already
 connectedETOOMANYREFSETIMEDOUTECONNREFUSEDhost is downEINPROGRESSstructure needs cleaning
 XENIX semaphores availableEREMOTEIOENOMEDIUMMEMEDIUMTYPEECANCELEDkey has expiredkey has
 been revokedkey was rejected by serviceowner died recoverablememory page has hardware
 errorillegal instructionbus erroruser signal broken terminatedstack faultstopped (tty
 input)urgent I/O conditionfile size limit exceededbad callunknown connection typeinvalid
 addressshort addresspacket too bigcertification path solicitation messageilnpv6 locator
 update messageextended echo requestinvalid connection implemented on destination
 unreachablerouter solicitationparameter problemtimestamp replyinvalid connectionnil
 headerreceived packet with unknown encryption level: %sunexpected encryption
 leveltQueueing ACK because the first packet should be acknowledged.tQueueing ACK because
 thereptpto (Initial)ptto (Application Data)negative bytes_in_flightinvalid packet number
 spacet await address validation any longer.tsetting loss timer packet %d (%s) to %s (in
 %s)Loss detection %s fired in PTO mode. PTO count: %dLimited by the number of tracked
 packets: tracking %d packets, maximum %d framesCryptoSetup: at this encryption level yet
 availabledecryption failedServerHelloCertificateRequestReceived %s message (%d bytes,
 encryption level: %s)unexpected handshake message: %dmismatching version. Got %d, expected
 %derror handling the handshake messageInstalled -RTT Read (using %s)Installed -RTT Read
 (using %s)Received -RTT key the serverInstalled -RTT Write (using %s)error creating new
 AES cipher: %sunexpected Retry integrity tag : %dfailed to RTTDropping key phase %d ahead
 of scheduled time. Drop was: %sunknown cipher suite %dInitiating key update to key phase
 %dIncreasing receive flow control window the connection to %d kBreceived inconsistent
 final offset stream %d (old: %d, new: %d bytes)duplicate stream datatoo many
 streamsputPacketBuffer called with packet of wrong size!invalid value
 Config.MaxIncomingStreamsreceived conflicting connection IDs sequence number %dexptected
 first connection ID to have sequence number Activating reading of ECN bits IPv4
 IPv6.Activating reading of ECN bits IPv6.received CRYPTO frame with unexpected encryption
 level: %sstream %d canceled with error code %d gap foundcannot different stateless on
 the same packet connfailed to determine receive buffer size: %wfailed to increase receive
 buffer size: %wfailed to increase receive buffer size (wanted: %d kiB, got %d kiB)Not
 adding connection ID %s, as it already exists.Adding connection IDs %s %s a new
 session.Replacing session connection ID %s with a closed session.received a packet with
 an unexpected connection ID %sSending stateless to %s (connection ID: %s). Token:
 %xan\unknown encryption levelPacketPacker BUG: packet too large (%d bytes, allowed %d
 bytes)unknown packet type: %sBUG: readPosInFrame (%d) > frame.DataLen (%d) in
 stream.Readserver closedDropping Version Negotiation packet.misrouted packet: %#v<-
 Received Initial packet.too short connection IDError sending Retry: %sSending a keep-alive
 PING to keep the connection alive.error parsing packet: %scoalesced packet has different
 destination connection ID: %s, expected %sDropping %s packet (%d bytes) that could be
 unpacked. Error: %sDropping (potentially) duplicate packet.Ignoring Retry, since we
 already received a packet.ignoring spoofed Retry. Integrity Tag doesn\<- Received
 Retry>Error parsing Version Negotiation packet: %sNo compatible QUIC version found.empty
 packetDestroying session: %sPeer closed session with error: %sRestoring Transport
 Parameters: %sreceived retry_source_connection_id, although Retry was performedt %s
 probe packet-> Sending packet %d (%d bytes) connection %s, %st queue undecryptable
 packets after handshake completiondeadline exceededpeer attempted to stream %dtried to
 incoming stream %d multiple tried to incoming stream %d multiple tried to unknown
 outgoing stream %dtried to unknown outgoing stream %dCERTIFICATE...:conn

```

closedUpdateConnStateba7b812-dad-d1-b4-c04fd430c8invalid UUID (got %d
bytes)urn:uuid:invalid UUID invalid UUID [ERR] yamux: Invalid protocol version: %dremote
yamux internal error[ERR] yamux: duplicate stream declaredoperation would block on IOkeep-
alive interval must be positivemax frame size must be larger than max stream buffer must
be positivemax stream buffer cannot be larger than websocket: limit exceededwebsocket:
bad message typewebsocket: bad handshakemalformed ws wss URLproxy: unknown scheme:
proxy: failed to parse port number: proxy: failed to greeting to SOCKS5 proxy at proxy:
SOCKS5 proxy at proxy: SOCKS5 proxy at proxy: failed to authentication request to SOCKS5
proxy at proxy: SOCKS5 proxy at proxy: failed to reply from SOCKS5 proxy at ( status)
(policy violation) (mandatory extension missing) (TLS handshake error)concurrent to
websocket connectioncontinuation after final message frameincorrect mask flagrepeated on
failed websocket connectionwebsocketSec-WebSocket-KeySec-WebSocket-ProtocolSec-WebSocket-
KeySec-WebSocket-Extensionswebsocket: duplicate header allowed: Sec-WebSocket-
Protocolpermessage-deflate; server_no_context_takeover;
client_no_context_takeoverpermessage-deflatesocks succeededconnection allowed by
rulesetTTL expirednil contextnetwork implementedusername/password authentication
failedtoo many authentication methods acceptable authentication methodsmethod is
allowedmux: duplicated route variable %qmux: missing name pattern in %qmux: unbalanced
braces in %qnil pointer to error encoderinvalid range: failed to overlapContent-Typeailed:
%s\nailed: %s\nre>up over\n%s sha256sum: %x\n<text/html; charset=utf-a style=><h5>now at:
%s<>\nContent-Typebytes *byteranges; boundary=If-None-MatchLast-ModifiedLast-
Modifiedinvalid rangeinvalid rangef\xff%xServerInitial
cpu.bmi1ermsrdrand).derivedderivedtls13 ->t%s
%#vEFAULTEEXISTEINVALENOTTYENOSPCEMPLINKERANGEENOLCKENOMSGEL3HLTELNRNGENOCIEBFONTENODATAENC
hptls13 serversuspendnullcc|)Removetimeoutsocks5: : : :
tcp4smuxsocks5//[&v//api/pre>\n/pre>\n%.fGB%.fEBW/W/EtagEtag//bytes=-/ cdnil EOF-
cICMP /SETENV1trueUSERTERMtrue

```

IOCs

wss://natsupport[.]net/api

References

- 1: <https://cloud.google.com/blog/topics/threat-intelligence/brickstorm-espionage-campaign>
- 2: <https://cloud.google.com/blog/topics/threat-intelligence/gostringungarbler-deobfuscating-strings-in-garbled-binaries>
- 3: <https://research.openanalysis.net/garble/go/obfuscation/strings/2023/08/03/garble.html>
- 4: <https://github.com/burrowers/garble>