BNB Chain 被駭全過程，BSC 還安全嗎？

原文：
https://www.btcc.com/zh-TW/academy/research-analysis/the-whole-process-of-bnb-chain-being-hacked

10 月 7 日，BNB Chain 的跨鏈橋（BSC Token Hub）遭到駭客攻擊， 有價值約 5 億美金的 BNB 被駭。由於涉及的金額較為龐大，並且涉及多個鏈之間的跨鏈，目前總損失金額仍未確定。

那麼，這起震動整個加密行業的重大安全事故是如何發生的呢？下文將為你一一進行分析。

相關報導：BNB 鏈跨鏈橋被駭，損失超 1 億美元，目前已恢復運作

為何駭客會盯上 BNB Chain？

台灣時間 10 月 7 日早上 6 點，BNB Chain 發推表示，由於跨鏈橋 BSC Token Hub 遭入侵，導致有額外的幣安幣（BNB）產生，因此 幣安智能鏈（BSC）將停機一段時間。同時幣安鏈將暫時暫停所有通過 BNB 鏈的存取款，直到有進一步的更新。

在不久後，BNB Chain 在另一推文表示，被提取資金約 7000 萬至8000 萬美元，已凍結 700 萬美元。

幣安（Binance）創辦人趙長鵬（CZ）也出面表示，由於 BNB Chain 跨鏈橋上的一個漏洞導致了額外的 BNB 幣的產生，已要求所有驗證者暫停 BNB Chain，這個問題現在得到了控制。同時，他強調用戶的資金是安全的，將會相應地提供進一步的更新。

由於跨鏈橋的複雜性以及累計的巨額財產，往往容易成為黑客攻擊的首要目標。

儘管區塊鏈在經過了一段時間的發展後，無論是區塊鏈項目方自己還是區塊鏈安全公司都對於安全的重視程度都高於了以往，但是跨鏈橋這種代碼複雜且含有鏈下部分的項目非常容易遭受攻擊。

跨鏈橋通常都是一些大項目，代碼量較多，多個環節的組合下就容易出現一些組合型漏洞，然而這些漏洞又是較為隱蔽的，容易被駭客所利用。跨鏈橋還有一個高危點就是鏈下安全，由於鏈下代碼一般與鏈上代碼分開審計，並且通常由項目方自己來保證安全，導致很多漏洞被忽視。

而這次駭客盯上了 幣安智能鏈（BSC）上的跨鏈橋（BSC Token Hub）。

BNB Chain 被駭全過程

---

10 月 7 號 零點 55 分，攻擊者於區塊高度 21955968 通過調用合約繳納 100 BNB 註冊成為 Relayer。

凌晨兩點半左右開始，駭客從 BNB Chain 的跨鏈橋系統合約中分兩次（2:26、4:43）共獲取了 200 萬枚 BNB。並將其中 90 萬枚 BNB 在 BNB Chain 上借貸協議 Venus 進行抵押，借出 6250 萬BUSD、5000 萬 USDT、3500 萬 USDC。

Beosin 安全團隊對其手法進行了詳細的解析：

幣安跨鏈橋 BSC Token Hub 在進行跨鏈交易驗證時，使用了一個特殊的預編譯合約用於驗證IAVL 樹。而該實現方式存在漏洞，該漏洞可能允許攻擊者偽造任意消息。因此，駭客透過以下過程成功竊取了資金。

1. 攻擊者先選取一個提交成功的區塊的哈希值（指定塊：110217401）
2. 然後構造一個攻擊載荷，作為驗證 IAVL 樹上的葉子節點
3. 在 IAVL 樹上添加一個任意的新葉子節點
4. 同時，添加一個空白內部節點以滿足實現證明
5. 調整第 3 步中添加的葉子節點，使得計算的根哈希等於第 1 步中選取的提交成功的正確根哈希
6. 最終構造出該特定區塊（110217401）的提款證明

目前，一些細節還要進一步推敲。

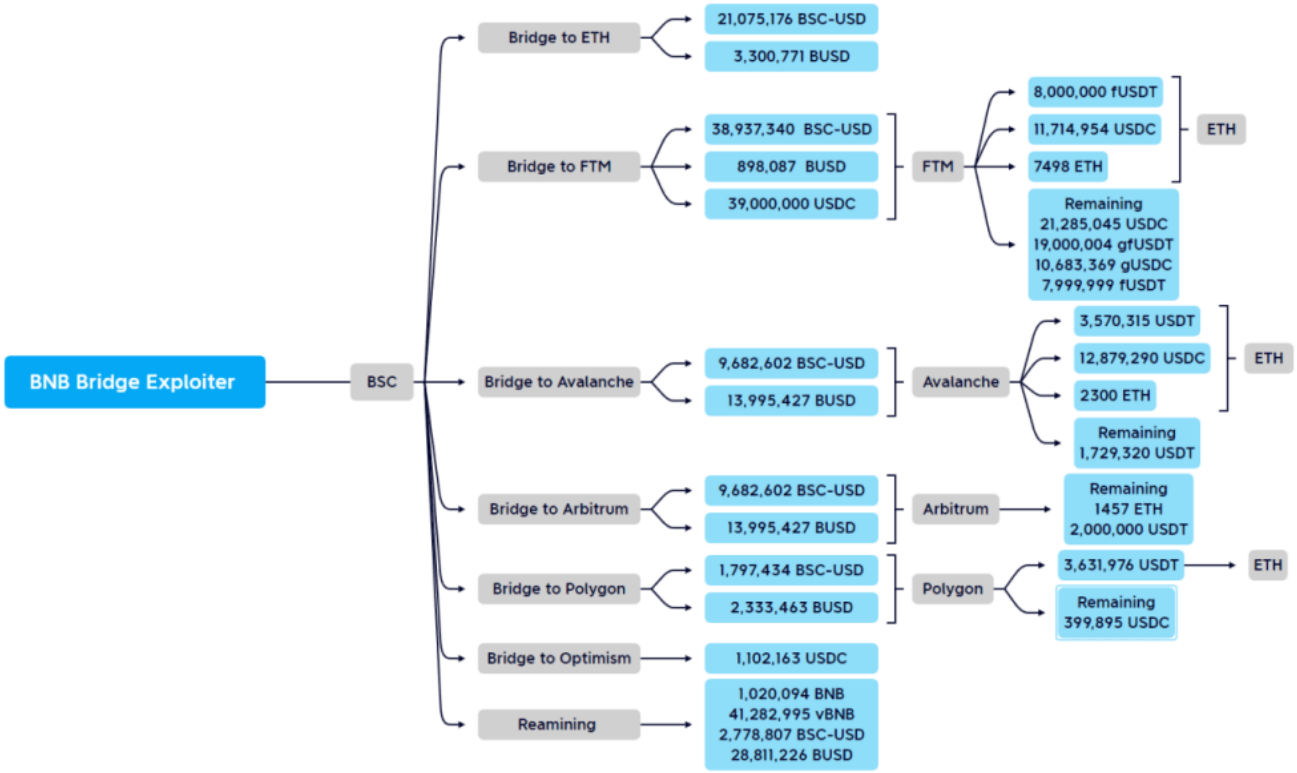
```
// Run executes a multi-store proof operation for a given value. It returns
// the root hash if the value matches all the store's commitID's hash or an
// error otherwise.
func (op MultiStoreProofOp) Run(args [][]byte) ([][]byte, error) {
        if len(args) != 1 {
                return nil, cmn.NewError("Value size is not 1")
        }

        value := args[0]
        root := op.Proof.ComputeRootHash()

        for _, si := range op.Proof.StoreInfos {
                if si.Name == string(op.key) {
                        if bytes.Equal(value, si.Core.CommitID.Hash) {
                                return [][]byte{root}, nil
                        }

                        return nil, cmn.NewError("hash mismatch for substore %v: %X vs %X", si.Name, si.Core.CommitID.Hash, value)
                }
        }

        return nil, cmn.NewError("key %v not found in multistore proof", op.key)
}
```

```
func (proof *RangeProof) computeRootHash() (rootHash []byte, treeEnd bool, err error) {
        if len(proof.Leaves) == 0 {
                return nil, false, cmn.ErrorWrap(ErrInvalidProof, "no leaves")
        }
        if len(proof.InnerNodes)+1 != len(proof.Leaves) {
                return nil, false, cmn.ErrorWrap(ErrInvalidProof, "InnerNodes vs Leaves length mismatch, leaves should be 1 more.")
        }

        // Start from the left path and prove each leaf.

        // shared across recursive calls
        var leaves = proof.Leaves
        var innersq = proof.InnerNodes
        var COMPUTEHASH func(path PathToLeaf, rightmost bool) (hash []byte, treeEnd bool, done bool, err error)

        // rightmost: is the root a rightmost child of the tree?
        // treeEnd: true iff the last leaf is the last item of the tree.
        // Returns the (possibly intermediate, possibly root) hash.
        COMPUTEHASH = func(path PathToLeaf, rightmost bool) (hash []byte, treeEnd bool, done bool, err error) {

                // Pop next leaf.
                nleaf, rleaves := leaves[0], leaves[1:]
                leaves = rleaves

                // Compute hash.
                hash = (pathWithLeaf{
                        Path: path,
                        Leaf: nleaf,
                }).computeRootHash()
```

BNB Chain 被駭資金

透過 Beosin Trace 對被盜資金進行追踪分析，Beosin 安全團隊發現總計有 1 億 4357 萬美元的被盜資金通過跨鏈進行轉移（含借貸）。被盜資金中有 7739 萬美元的資金通過各種跨鏈轉入了以太坊，5896 萬美元的資金留存在 FTM 鏈中（含各種 gUSDT），400 萬美元的資金在 Arbitrum 鏈中，172 萬美元的資金在 Avalanche 鏈中，40 萬美元的資金在 Polygon 和 110 萬美元在 Optimism。

具體資金流向如下圖所示：



BNB Chain 被駭資金流向

BNB Chain 還安全嗎？

---

10 月 7 日 9 點半左右，BNB Chain 官方在社媒平台上發文表示，已要求 BNB Chain 節點驗證者在未來幾個小時內與其聯繫，以便可以計劃進行節點升級。

到了下午一點，BNB Chain 發推稱，已發布 BSC v1.1.15 版本，BSC 驗證者正在協調，以尋求在 1 小時內恢復 BNB 智能鏈（BSC）。新版本將阻止黑客帳戶相關活動。BNB 信標鏈和 BNB 智能鏈之間的原生跨鏈通信已禁用。並且，官方要求所有節點運營者嘗試升級至上述版本。驗證者和社區將討論進一步升級以完全解決此問題。

下午三點左右，BNB Chain 發推稱，BNB 智能鏈（BSC）20 多分鐘前開始良好運行。驗證者正在確認他們的狀態，社區基礎設施也在升級。此外，BscScan 數據顯示，BNB Chain 網絡已恢復出塊。

Beosin 安全團隊監測顯示，重啟之後，當前 BSC 節點程序將通過黑名單與暫停 iavlMerkleProofValidate 功能的方式阻止被盜資金流動與潛在的攻擊。

但能否確保 BNB Chain 不會再發生此類安全事故，還需要團隊不斷完善項目安全。

結語

總的來說，以往的跨鏈橋攻擊多為通過線下漏洞或者是私鑰洩露等方式，而本次的 BNB Chain 攻擊卻是通過的構造特定的根哈希來構造出特定區塊的提款證明，從而使攻擊成立，攻擊難度比較大，並且數額較以往來說也比較高。但幣安也在發現了狀況後及時進行了阻攔，將損失降低。

這也提醒了各位投資者，沒有任何項目是絕對安全的。尤其是區塊鏈的發展時間並不長，會出現許多意想不到的問題，因此，建議您在保存資產時不要全部儲存在一個錢包和項目中。
