APPENDIX

A. Proof of Proposition III.3

Proposition III.3 reveals that one-round reverse engineering by Equation 2 can only invert a trigger $\boldsymbol{\delta}$ in the target trigger set $\mathcal{T}$ with a limited hyper-parameter λ carefully adjusted by developers. To analyze how one-round reverse engineering helps to find the injected trigger, we use the following assumptions for ease of proof.

Assumption A.1. For any given encoder f, there always exists a solution for Equation 2, which is $\boldsymbol{\delta} \in \mathcal{Q}$. We have $\forall \boldsymbol{\delta} \in \mathcal{Q}, psim(X, \boldsymbol{\delta}) = 1$ or $psim(X, \mathcal{Q}) = 1$ for simplicity.

Assumption A.2. For any poisoned encoder f', there exist two solutions for Equation 2: $\boldsymbol{\delta} \in \mathcal{T}$ and $\boldsymbol{\delta} \in \mathcal{Q}$.

Assumption A.1 tells that we can always find an extremely large trigger on an encoder to have the maximum pair-wise similarity. It can be easily achieved by setting the trigger mask to all ones. Assumption A.2 shows that for backdoored encoders, we can also find small triggers belonging to the target trigger set. Note that it is fairly easy to find small triggers on classifiers, even on clean classifiers. The two assumptions here may not hold in supervised learning. Figure 7 showcases the empirical evidence of this assumption. Based on these assumptions, we have the following proof.

Proof. According to Assumption A.2, there are two solutions for Equation 2 to obtain the trigger: $\boldsymbol{\delta} \in \mathcal{T}$ and $\boldsymbol{\delta} \in \mathcal{Q}$. As shown in Assumption A.1, it is much easier to obtain $\boldsymbol{\delta} \in \mathcal{Q}$ as one can simply set mask $\boldsymbol{m} = 1$. It is hence necessary to have the regularization term $||\boldsymbol{\delta}||_1$ to avoid the trivial and yet useless solution. We denote the optimization objectives for the two solutions as follows.

$$\mathcal{L}_{\mathcal{Q}} = -psim(X, \boldsymbol{\delta}_{\mathcal{Q}}) + \lambda||\boldsymbol{\delta}_{\mathcal{Q}}||_1$$
$$\mathcal{L}_{\mathcal{T}} = -psim(X, \boldsymbol{\delta}_{\mathcal{T}}) + \lambda||\boldsymbol{\delta}_{\mathcal{T}}||_1$$

We aim to demonstrate the following is smaller than zero.

$$\mathcal{L}_{\mathcal{T}} - \mathcal{L}_{\mathcal{Q}} = \mathcal{P}(X, \boldsymbol{\delta}_{\mathcal{Q}}, \boldsymbol{\delta}_{\mathcal{T}}) + \lambda \cdot (||\boldsymbol{\delta}_{\mathcal{T}}||_1 - ||\boldsymbol{\delta}_{\mathcal{Q}}||_1).$$
$$\big[psim(X, \boldsymbol{\delta}_{\mathcal{Q}}) - psim(X, \boldsymbol{\delta}_{\mathcal{T}})\big] + \lambda \cdot (||\boldsymbol{\delta}_{\mathcal{T}}||_1 - ||\boldsymbol{\delta}_{\mathcal{Q}}||_1).$$

From Assumption A.1, we have $psim(X, \boldsymbol{\delta}_{\mathcal{Q}}) = 1$. Since the triggers in $\mathcal{T}$ are at most as good as the injected trigger, their pair-wise similarity $psim(X, \boldsymbol{\delta}_{\mathcal{T}})$ is at best 1. This means that the first term $\big[psim(X, \boldsymbol{\delta}_{\mathcal{Q}}) - psim(X, \boldsymbol{\delta}_{\mathcal{T}})\big]$ is close to 0. According to Definition III.2, $\forall \boldsymbol{\delta}_i \in \mathcal{T}, \forall \boldsymbol{\delta}_j \in \mathcal{Q}, ||\boldsymbol{\delta}_i||_1 \ll ||\boldsymbol{\delta}_j||_1$. Thus, the second term $\lambda \cdot (||\boldsymbol{\delta}_{\mathcal{T}}||_1 - ||\boldsymbol{\delta}_{\mathcal{Q}}||_1) \ll 0$. We hence have $\mathcal{L}_{\mathcal{T}} - \mathcal{L}_{\mathcal{Q}} < 0$, meaning one-round reverse engineering has the ability to find a trigger in the target trigger set $\mathcal{T}$. However, it is highly dependent on the hyper-parameter λ. A large value leads to a particularly small trigger and fails to satisfy the pair-wise similarity requirement; a small value causes the inversion to produce triggers in set $\mathcal{Q}$. Without knowledge of the value $(||\boldsymbol{\delta}_{\mathcal{T}}||_1 - ||\boldsymbol{\delta}_{\mathcal{Q}}||_1)$, there is no guidance for choosing the hyper-parameter λ and the hyper-parameter tuning is based on manual experience. □

B. Proof of Theorem III.4:Inevitable Trigger Size Growth

As introduced in Algorithm 2, multi-round reverse engineering first generates a trigger using reverse engineering and then trains the encoder to unlearn this trigger, as in Equation 3. It then repeats the above procedure for multiple rounds, which helps to approach the injected trigger $\boldsymbol{\delta}^*$ in $\mathcal{T}$. We denote this process as *model hardening*, and trigger size growth is then observed. To simplify, we introduce the definition of $f^(k)$ and the assumption of hardening monotonicity at first.

Definition A.3 ($f^{(k)}$)**.** $f^{(k)}$ is defined as the k-th round encoder during model hardening, where $f^{(0)}$ is the initial backdoored encoder and $k \leq K$ (the total hardening rounds).

Assumption A.4 (Hardening Monotonicity)**.** $\forall 0 \leq k_i \leq k_j \leq K$, the number of triggers in the target trigger set $\mathcal{T}$ that can attack $f^{(k_j)}$ is smaller than that on $f^{(k_i)}$.

As the model hardening process dispels triggers that can cause misclassification, Assumption A.4 illustrates that model hardening improves model's robustness monotonically. Based on this, we will show that the inverted trigger size grows inevitably. First, we have the following proposition that the trigger inverted on the encoder in later rounds is always larger than that on the encoder in early rounds.

Proposition A.5. *$\forall 0 \leq k_i \leq k_j \leq K$, $||\boldsymbol{\delta}_{k_i}||_1 \leq ||\boldsymbol{\delta}_{k_j}||_1$, where $\boldsymbol{\delta}_k$ is the trigger inverted on k-th round encoder $f^{(k)}$.*

Proof. We prove the proposition by induction.
Case 1: We prove the proposition is true when $K = 1$. When $k_i = k_j, ||\boldsymbol{\delta}_{k_i}||_1 = ||\boldsymbol{\delta}_{k_j}||_1$ and hence the proposition is true. When $k_i < k_j$, we have $k_i = 0, k_j = 1$. In this case, $\boldsymbol{\delta}_{k_i} \in \mathcal{T}$ and $\boldsymbol{\delta}_{k_j} \in \mathcal{Q}$. Therefore, $||\boldsymbol{\delta}_{k_i}||_1 \leq ||\boldsymbol{\delta}_{k_j}||_1$ as per Definition III.2.
Case 2: Assume the proposition is true when $K = n$, we aim to prove that it is also true when $K = n + 1$. According to Assumption A.4, $f^{(n+1)}$ is closer to a completely clean encoder compared to $f^{(n)}$. Based on the hardening monotonicity, we may assume that $f^{(n)}$ is backdoored while $f^{(n+1)}$ is clean. Therefore, we have $||\boldsymbol{\delta}_n||_1 \leq ||\boldsymbol{\delta}_{n+1}||_1$ and the proposition is true when $K = n + 1$.
The original proposition is hence proved. □

There are two loss terms in Equation 2: pair-wise similarity and size regularization. Based on Assumption A.4, for a trigger $\boldsymbol{\delta}^{(k)}$ inverted on $f^{(k)}$, its pair-wise similarity on $f^{(k)}$ is smaller than that on $f^{(k+1)}$. Thus, the hyper-parameter λ needs to be reduced to satisfy the first loss term, leading to the growth of the trigger size. Figure 7 illustrates the empirical evidence.

C. Proof of Theorem III.5:Paradox of Security and Utility

Proof. We prove the theorem by contradiction. Assume $\mathcal{M}^{(K)}$ is the final backdoor-removed encoder with preserved model utility. According to Theorem III.4, the inverted trigger grows slowly starting from a small trigger until it covers $\mathcal{T}$ and $\mathcal{T}^+$. This means $\exists n < K$ and $K - n \geq 2$, $\boldsymbol{\delta}_n \in \mathcal{T}$ and $\boldsymbol{\delta}_{n+1} \in \mathcal{T}^+ - \mathcal{T}$ (i.e., $\mathcal{Q}$), where $\boldsymbol{\delta}_n$ denotes the inverted

trigger on encoder $\mathcal{M}^{(n)}$. During round $K-n$, the adversarial training aims to unlearn the inverted triggers $\boldsymbol{\delta} \notin \mathcal{T}$ (i.e., $\boldsymbol{\delta} \in \mathcal{Q}$). According to Definition III.2, the triggers in $\mathcal{Q}$ have arbitrary patterns. In addition, based on Proposition A.5 and Theorem III.4, with the increase of the training round (after n), the trigger quality degrades as it is moving further away from the exact injected trigger. With $K-n \geq 2$, the encoder is trained on multiple rounds of arbitrary inputs and low-quality triggers, inevitably leading to utility degradation. This contradicts the initial assumption. □

D. *Proof of Theorem III.6*

Proof. According to Theorem III.4, the inverted trigger grows slowly starting from a small trigger until it covers $\mathcal{T}$ and $\mathcal{T}^+$. This means $\exists\, \boldsymbol{\delta}_k, \boldsymbol{\delta}_{k+1} \in \mathcal{T}^+, \boldsymbol{\delta}_k \in \mathcal{T}$ and $\boldsymbol{\delta}_{k+1} \in \mathcal{T}^+ - \mathcal{T}$ (i.e., $\mathcal{Q}$), where $\boldsymbol{\delta}_k$ denotes the inverted trigger on encoder $\mathcal{M}^{(k)}$. In other words, the injected trigger $\boldsymbol{\delta}^*$ is between $\boldsymbol{\delta}_k$ and $\boldsymbol{\delta}_{k+1}$, which is the optimal solution. Based on the partial order defined in Definition III.1, we have $||\boldsymbol{\delta}_k||_1 \leq ||\boldsymbol{\delta}^*||_1 < ||\boldsymbol{\delta}_{k+1}||_1$. With a small enough learning rate, the multi-round trigger inversion produces triggers with continuous sizes. Since there is no member in $\mathcal{Q}$ that is between $\boldsymbol{\delta}_k$ and $\boldsymbol{\delta}_{k+1}$, $\boldsymbol{\delta}_k$ is the optimal trigger $\boldsymbol{\delta}^*$. □