IX. Supplementary Document

A. More Results on Validating Decomposed Clean Inputs and Triggers

We provide additional results in Table XIII to validate decomposed clean inputs and triggers in Section III-A. The first column denotes the dataset, the second column the network structure, and the third column the backdoor attack type. There are two large column blocks, presenting the decomposition results of clean inputs and triggers, respectively. In each block, we show the L^1 distance, PSNR, SSIM, and ACC/ASR. For L^1, PSNR, and SSIM, the first column denotes the difference between trojaned images and their clean counterparts. The second column shows the difference between original clean images and decomposed images by Februus (Fb) and the third column the difference by BEAGLE (Bg). Each entry presents the average value for the given 10 trojaned images and 100 additional clean test images. For ACC/ASR, the first column is for Februus and the second for *Beagle*. Each entry in the ACC columns denotes the average clean accuracy for the decomposed clean images from the given 10 trojaned samples. As the 100 validation clean images have been used in optimization, we use the images from the test set and stamp the decomposed trigger to calculate the ASR. The last row in Table XIII shows the average results. Better results between Februus and BEAGLE are highlighted with the bold font.

Validating Decomposed Triggers Across Datasets/Models. In the previous experiments, the triggers are decomposed and validated on the same dataset and model. BEAGLE can effectively extract the trigger with high visual quality and classification accuracy. In real-world scenarios, the trojaned model and its dataset that are available for forensics might be different from the subject model/dataset. Here, we study the performance of BEAGLE in decomposing triggers across different datasets and model architectures. Particularly, we use the same injected trigger to poison a VGG-11 model on CIFAR-10 and a ResNet-18 model on GTSRB. We apply BEAGLE to extract the trigger from the trojaned VGG-11 model and then test it on the ResNet-18 (CV-GR) and vice versa. Figure 18 shows the results. The x-axis denotes the different backdoor attacks, and the y-axis the ASR of the decomposed trigger on the other model. Observe that almost all the ASRs are high for 10 different attacks, meaning the decomposed triggers are highly effective even when applied on a different dataset/model. This demonstrates the decomposition of BEAGLE is general and does not overfit on specific datasets/models.

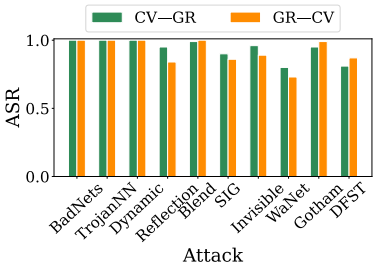


Fig. 18: Validation on decomposed triggers across datasets/models

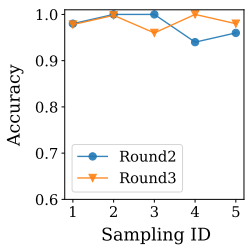


Fig. 19: Summarization of different attacks in TrojAI

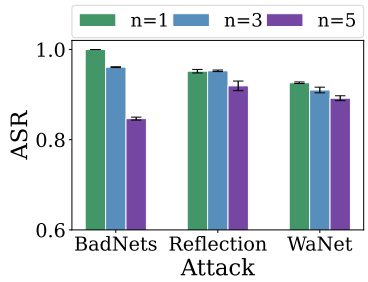


Fig. 20: Impact of including naturally miclassified images

B. More Results of Evaluation on Attack Clustering and Summarization

In Section IV-D, we evaluate BEAGLE's attack clustering and summarization performance on CIFAR-10 for seven backdoor attacks. Here, we evaluate on TrojAI rounds 2 and 3. Specifically, we randomly sample 50 trojaned models (out of 1000 total) that are injected with polygon or (one of the 5 types of) filter triggers and apply BEAGLE to identify the attack types. We conduct the experiment for five times (trials) and report the results in Figure 19. The x-axis denotes the trial ID and the y-axis the accuracy of correctly identifying the attack type. Observe that the recognition accuracies are nearly 100% for both rounds across the five random trials. We make use of four well-known clustering methods: (1) K-means [59] equipped with Silhouette Score [73], abbreviated as K-means-S; (2) K-means [59] equipped with Elbow method [91], abbreviated as K-means-E; (3) Gaussian Mixture Model (GMM) [72] equipped with Silhouette Score [73], abbreviated as GMM-S; and (4) DBSCAN [19]. Since K-means and GMM require to pre-set the number of clusters, we test different numbers of clusters, and leverage the Silhouette Score and Elbow methods to determine the optimal number (with the highest score). DBSCAN can automatically determine the optimal number of clusters. Their clustering results have negligible differences (all having 4-6 clusters). The downstream synthesized scanners have similar performance too (see Table XVIII).

C. Impact of Attack Sample Bias

As BEAGLE leverages a small set of inputs (clean and trojaned) and trojaned models, we study the impact of sampling biases in such data. For sampled inputs, we include a few naturally misclassified inputs without any injected backdoors. This simulates the real world scenario where classification models do not usually achieve 100% accuracy. For sampled models, we intentionally introduce biases to the number of trojaned models with different attack types.

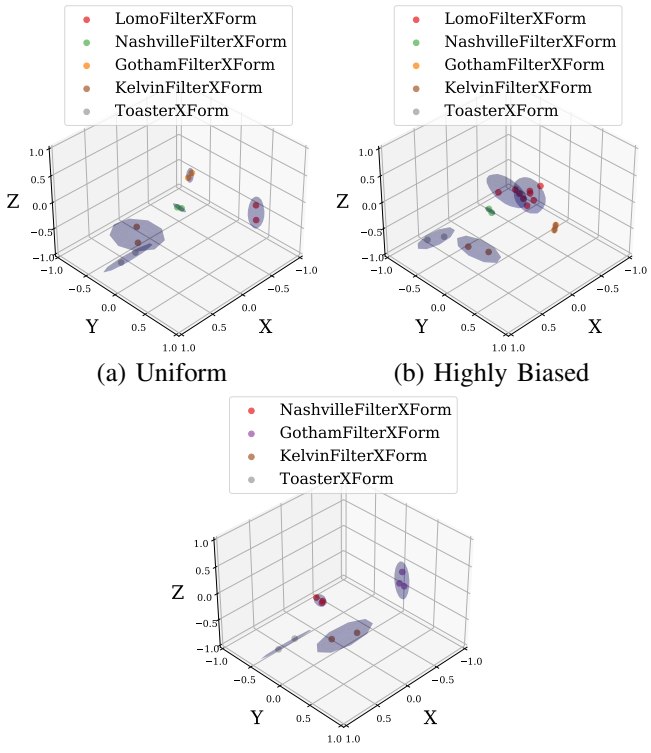
Attack Samples including Naturally Misclassified Images. We mix different numbers of misclassified images with trojaned samples and then measure the ASRs of the decomposed triggers (extracted from these inputs) on the test set. A high ASR means the decomposed trigger is effectively decomposed and hence useful to synthesize scanners. We conduct the experiments on CIFAR-10 with VGG-11 and three attacks: BadNets, Reflection, and WaNet. We randomly select 1, 3 and 5 misclassified images and mix them with 9, 7 and 5 trojaned samples, respectively. We run the experiment of each setting for five times and show the results in Figure 20. The x-axis denotes different attacks and the y-axis the ASR. Each bar shows the ASR of the decomposed trigger and the whisker denotes the variance.

TABLE XIII: Validation on decomposed clean images and triggers

DS	Nk	Attack	Decomposed Clean Images										Decomposed Trigger									
			L1 ↓			PSNR ↑			SSIM ↑			ACC ↑	L1 ↓			PSNR ↑			SSIM ↑			ASR ↑
			$x \oplus t$	Fb.	Bg.	$x \oplus t$	Fb.	Bg.	$x \oplus t$	Fb.	Bg.	Fb. Bg.	x	Fb.	Bg.	x	Fb.	Bg.	x	Fb.	Bg.	Fb.	Bg.
ImageNet	VGG-16	BadNets	0.037	0.128	**0.088**	22.65	22.86	**26.93**	0.95	0.64	**0.75**	1.0 **1.0**	0.038	**0.023**	0.042	22.51	24.30	**24.37**	0.95	**0.97**	0.91	0.63 **1.00**
		TrojNN	0.065	0.157	**0.083**	19.41	21.60	**28.69**	0.82	0.54	**0.75**	0.5 **1.0**	0.067	0.077	**0.044**	19.17	19.15	**23.60**	0.84	0.84	**0.85**	0.97 **1.00**
		Invisible	0.075	**0.151**	0.166	29.46	**23.26**	23.04	0.89	0.56	**0.57**	0.1 **1.0**	0.093	0.222	**0.140**	27.43	17.48	**24.08**	0.90	**0.77**	0.76	0.13 **0.95**
		WaNet	0.024	**0.112**	0.134	38.70	**25.36**	25.01	0.98	0.67	**0.67**	0.7 **1.0**	0.034	0.223	**0.217**	33.18	16.83	**20.39**	0.97	**0.72**	0.70	0.10 **0.91**
		Gotham	0.409	0.443	**0.214**	15.25	14.49	**20.70**	0.77	0.50	**0.59**	0.0 **1.0**	0.428	0.643	**0.200**	15.05	11.62	**20.35**	0.76	0.40	**0.75**	0.21 **0.99**
CelebA	ResNet-18	BadNets	0.047	0.090	**0.036**	20.60	20.55	**32.59**	0.94	0.87	**0.94**	1.0 **1.0**	0.045	**0.027**	0.079	20.94	**24.07**	19.88	0.94	**0.96**	0.94	0.95 **1.00**
		TrojNN	0.062	0.113	**0.041**	20.28	21.47	**31.06**	0.81	0.75	**0.94**	0.0 **1.0**	0.061	**0.060**	0.078	20.37	20.77	**20.77**	0.81	0.83	**0.86**	0.78 **1.00**
		Reflection	0.469	0.472	**0.243**	13.98	13.92	**19.83**	0.43	0.40	**0.62**	0.0 **1.0**	0.443	0.450	**0.177**	14.71	14.51	**21.17**	0.49	0.46	**0.72**	0.23 **0.98**
		SIG	0.380	0.384	**0.209**	16.43	16.24	**21.14**	0.67	0.59	**0.73**	0.0 **1.0**	0.390	0.374	**0.167**	16.09	16.41	**23.25**	0.64	0.56	**0.85**	0.20 **0.93**
		Blend	0.293	0.306	**0.254**	18.77	18.48	**19.09**	0.56	0.52	**0.62**	0.4 **1.0**	0.248	0.268	**0.175**	20.07	19.45	**20.58**	0.63	0.59	**0.73**	0.44 **1.00**
CIFAR-10	VGG-11	BadNets	0.040	0.082	**0.03**	22.04	21.69	**31.14**	0.95	0.93	**0.97**	0.8 **1.0**	0.038	0.028	**0.016**	22.75	25.94	**28.87**	0.95	**0.98**	0.96	1.00 **1.00**
		TrojNN	0.070	0.156	**0.025**	18.96	19.16	**33.41**	0.75	0.76	**0.98**	0.1 **1.0**	0.067	0.221	**0.008**	19.37	16.53	**30.58**	0.75	0.70	**0.98**	0.84 **1.00**
		Dynamic	0.021	0.061	**0.054**	25.83	**26.28**	25.83	0.95	0.94	**0.94**	1.0 **1.0**	0.022	0.150	**0.081**	25.61	18.02	**20.18**	0.95	0.77	**0.81**	0.99 **1.00**
		Reflection	0.436	0.441	**0.212**	15.07	14.96	**21.08**	0.53	0.49	**0.81**	0.0 **1.0**	0.446	0.430	**0.164**	14.93	15.35	**22.92**	0.56	0.52	**0.89**	0.17 **0.98**
		SIG	0.274	0.297	**0.191**	19.82	18.58	**21.89**	0.59	0.51	**0.84**	0.3 **1.0**	0.272	0.270	**0.077**	18.57	16.48	**27.73**	0.67	0.52	**0.96**	0.44 **0.94**
		Blend	0.183	0.220	**0.196**	22.60	19.63	**21.92**	0.81	0.66	**0.84**	0.2 **1.0**	0.187	0.291	**0.118**	22.43	17.85	**26.08**	0.82	0.53	**0.94**	0.21 **1.00**
		Invisible	0.061	**0.080**	0.091	31.07	**28.38**	28.29	0.97	0.94	**0.96**	0.1 **1.0**	0.061	0.123	**0.099**	30.93	21.62	**27.12**	0.96	0.82	**0.88**	0.10 **0.92**
		WaNet	0.057	**0.102**	0.116	29.96	23.90	**26.20**	0.97	0.92	**0.95**	0.0 **1.0**	0.059	0.337	**0.101**	29.56	15.41	**27.68**	0.96	0.50	**0.97**	0.12 **0.90**
		Gotham	0.457	0.503	**0.183**	14.84	13.59	**22.49**	0.80	0.64	**0.91**	0.8 **1.0**	0.488	0.654	**0.185**	14.45	11.49	**20.96**	0.78	0.34	**0.90**	0.29 **0.97**
		DFST	0.593	0.623	**0.454**	13.48	13.01	**15.83**	0.65	0.59	**0.74**	0.2 **1.0**	0.632	0.780	**0.383**	12.82	10.51	**16.41**	0.61	0.18	**0.68**	0.16 **0.92**
GTSRB	ResNet-18	BadNets	0.043	0.087	**0.044**	21.88	**27.67**	26.39	0.94	0.92	**0.95**	1.0 **1.0**	0.041	0.237	**0.013**	22.14	17.02	**29.81**	0.93	0.62	**0.98**	1.00 **1.00**
		TrojNN	0.072	0.127	**0.020**	18.81	19.40	**37.94**	0.71	0.74	**0.99**	0.6 **1.0**	0.070	0.076	**0.001**	18.81	20.45	**49.01**	0.69	0.82	**1.00**	1.00 **1.00**
		Dynamic	0.023	0.078	**0.034**	25.19	27.73	**30.21**	0.93	0.89	**0.95**	1.0 **1.0**	0.026	0.640	**0.065**	24.63	11.64	**21.16**	0.94	0.16	**0.84**	1.00 **1.00**
		Reflection	0.565	0.567	**0.315**	12.70	12.64	**17.60**	0.32	0.27	**0.59**	0.0 **1.0**	0.547	0.458	**0.277**	12.88	13.64	**18.39**	0.22	0.37	**0.77**	0.14 **0.97**
		SIG	0.275	0.306	**0.188**	19.18	17.91	**22.57**	0.47	0.37	**0.72**	0.7 **1.0**	0.275	0.287	**0.081**	19.09	18.65	**28.36**	0.37	0.59	**0.96**	0.15 **0.89**
		Blend	0.208	0.220	**0.213**	21.47	20.76	**20.79**	0.71	0.71	**0.72**	0.2 **1.0**	0.206	0.308	**0.126**	21.67	17.79	**25.39**	0.68	0.54	**0.90**	0.16 **1.00**
		Invisible	0.058	**0.123**	0.152	31.70	23.78	**24.71**	0.94	0.80	**0.92**	0.0 **1.0**	0.055	**0.123**	0.149	32.18	17.02	**24.25**	0.94	0.50	**0.92**	0.43 **0.96**
		WaNet	0.048	**0.088**	0.094	33.11	26.10	**28.38**	0.97	0.88	**0.95**	0.0 **1.0**	0.042	0.152	**0.072**	33.42	20.14	**30.52**	0.97	0.68	**0.96**	0.42 **0.97**
		Gotham	0.427	0.450	**0.222**	14.68	14.18	**21.08**	0.74	0.68	**0.87**	0.8 **1.0**	0.446	0.645	**0.153**	14.48	11.83	**23.45**	0.77	0.32	**0.81**	0.22 **1.00**
		DFST	0.597	0.592	**0.582**	13.69	**13.80**	13.88	0.62	0.59	**0.62**	0.0 **1.0**	0.709	0.826	**0.788**	11.58	9.99	**10.47**	0.48	0.22	**0.51**	0.42 **0.97**
Average			0.202	0.245	**0.155**	21.29	19.97	**25.12**	0.78	0.69	**0.82**	0.38 **1.0**	0.207	0.316	**0.147**	21.02	16.90	**23.66**	0.77	0.58	**0.83**	0.45 **0.96**

Observe that the ASR decreases with the increase of the number of misclassified images, which is expected. But with half of the inputs being naturally misclassified ones (5 out of 10), the ASR of the decomposed trigger is still around 90%, delineating the robustness of BEAGLE in attack decomposition. The reason is that the natural samples unlikely have consistent misclassification-inducing features and hence they hardly affect the decomposition results.

Biases in Attack Types. We leverage five filter attacks in TrojAI round 3 and study three different sampling settings: (1) *uniform* sampling that samples the same number of trojaned models from each attack type; (2) *highly-biased* sampling that samples a large number of models for some attack type; (3) *missing* sampling that does not include some attack type. We then apply BEAGLE to partition the sampled trojaned models. Figure 21 shows the partitioning results using K-means-S. Each point denotes the reduced attack features for each trojaned model. Observe that in Figure 21 (b) with the majority of models trojaned by the Lomo filter, BEAGLE can still accurately partition them. Similar observation can be made in Figure 21 (c). We also evaluate the detection accuracy of the synthesized scanners and report the results in Table XIV. Observe that the scanning accuracies are more than 90% for the uniform and highly-biased cases. The accuracy for the missing attack setting is slightly lower (87%) as BEAGLE could not synthesize the scanner for the unseen attack.



(a) Uniform (b) Highly Biased

(c) Missing

Fig. 21: Partitioning results under different sampling settings

D. Adaptive Attack

We study three attack scenarios where the adversary has the knowledge of BEAGLE.

Robust Trigger Injection. Particularly, the adversary first trains a trojaned model that has high clean accuracy and ASR. He then follows BEAGLE to extract the injected trigger and stamps the

TABLE XIV: Scanning accuracy under different sampling settings

Sampling Setting	Uniform	Biased	Missing
Number of clusters	5	5	4
Accuracy	0.93	0.92	0.87

TABLE XV: Results on adaptive attack

Backdoor	ACC	ASR	Decomposed Clean Images				Decomposed Trigger			
			L1↓	PSNR↑	SSIM↑	ACC↑	L1↓	PSNR↑	SSIM↑	ASR↑
BadNets	0.919	1.000	0.030	31.14	0.97	1.0	0.016	28.87	0.96	1.00
	0.911	1.000	0.029	31.51	0.98	1.0	0.018	28.19	0.96	0.99
TrojNN	0.917	1.000	0.025	33.41	0.98	1.0	0.008	30.58	0.98	1.00
	0.908	0.997	0.024	34.91	0.99	1.0	0.004	34.86	0.99	1.00
Dynamic	0.919	1.000	0.054	25.83	0.94	1.0	0.081	20.18	0.81	1.00
	0.911	1.000	0.056	26.20	0.94	1.0	0.088	19.88	0.79	1.00
Reflection	0.918	0.991	0.212	21.08	0.81	1.0	0.164	22.92	0.89	0.98
	0.917	**0.850**	0.213	21.03	0.81	1.0	0.169	22.53	0.88	**0.81**
SIG	0.914	0.952	0.191	21.89	0.84	1.0	0.077	27.73	0.96	0.94
	0.908	**0.566**	0.198	21.53	0.84	1.0	0.090	25.71	0.93	**0.21**
Blend	0.920	1.000	0.196	21.92	0.84	1.0	0.118	26.08	0.94	1.00
	0.909	1.000	0.196	21.84	0.84	1.0	0.118	26.02	0.93	0.99
Invisible	0.918	1.000	0.091	28.29	0.96	1.0	0.099	27.12	0.88	0.92
	0.907	0.998	0.089	28.39	0.96	1.0	0.089	28.14	0.91	0.89
WaNet	0.908	0.989	0.116	26.20	0.95	1.0	0.101	27.68	0.97	0.90
	0.882	**0.639**	0.117	26.14	0.95	1.0	0.097	27.91	0.96	**0.61**
Gotham	0.913	1.000	0.183	22.49	0.91	1.0	0.185	20.96	0.90	0.97
	0.911	0.991	0.182	22.53	0.91	1.0	0.188	20.70	0.89	0.89
DFST	0.889	0.996	0.454	15.83	0.74	1.0	0.383	16.41	0.68	0.92
	0.878	0.996	0.425	16.22	0.75	1.0	0.383	16.53	0.67	0.77

decomposed trigger on clean samples with their original labels. Poisoned in such a way, the trojaned model is not sensitive to the decomposed trigger and has a more robust injected backdoor. We apply BEAGLE on those trojaned models and present the results in Table XV. Column 1 denotes the attacks. Columns 2-3 show the clean accuracy (ACC) and attack success rate (ASR). The remaining columns show the results on visual quality and classification accuracy as introduced in Section IV-C. For each attack in the table, we report the results on the original trojaned models and on the models by the adaptive attack in the first and the second rows, respectively. Observe that for most attacks, the adaptive attack induces slight degradation on clean accuracy and ASR. For those cases, BEAGLE can still decompose high quality clean images and triggers. In contrast for Reflection, SIG, and WaNet, the decomposed triggers' ASRs have nontrivial degradation, indicating that BEAGLE becomes less effective. However, observe that the ASRs of the injected triggers also degrade a lot, suggesting the adaptive attack is ineffective either. The reason is that the decomposed triggers are so similar to the injected ones that they cancel each others out.

Mixing Targeted Adversarial Examples. The adversary in this scenario mixes targeted adversarial examples (i.e., adversarial examples misclassified to the target class) with the collected trojaned instances to affect the decomposition process of BEAGLE. There are two cases. First given a clean model, the adversary provides 10 targeted adversarial examples. In this case, we evaluate whether BEAGLE can still correctly classify clean models. We generate a set of targeted adversarial samples using PGD with a reasonable bound $L_\infty = 16/256$ and feed them to BEAGLE. BEAGLE can only decompose a trigger that achieves 69.7% ASR on the clean validation samples. We use the synthesized scanner to scan 10 clean models, and the scanning accuracy is 100%, which means no model is considered trojaned. The reason is that targeted adversarial attack, not like backdoor attack, does not have a universal secret trigger. Therefore, BEAGLE cannot summarize a trigger with a high ASR, and hence the synthesized scanner will not classify clean models as trojaned. In the second case, the adversary mixes targeted adversarial examples with real trojaned samples to affect BEAGLE's decomposition process, which is similarly to that of mixing naturally misclassified samples IX-C. We conduct experiments on CIFAR-10 with VGG-11 and three attacks, BadNets, Reflection, and WaNet. We randomly mix 1, 3, and 5 targeted adversarial examples with 9, 7, and 5 real trojaned samples respectively. Table XVI shows the results where the numbers in the table denote the ASR of decomposed trigger. Observe that the ASR slightly decreases with the increase of the number of targeted adversarial examples as expected. However, even when half of the given samples are adversarial examples, BEAGLE can still synthesize effective triggers.

TABLE XVI: Impact of mixing targeted adversarial examples.

Number	BadNets	Reflection	WaNet
1	0.999	0.959	0.921
3	0.978	0.939	0.865
5	0.859	0.925	0.809

Injecting Multiple Trojans. The adversary may inject multiple backdoors into a subject model to affect the decomposition process of BEAGLE. Although this scenario is beyond our threat model I, where we assume all the trojaned inputs in an attack instance used in forensics are exploiting the same backdoor, we conduct the experiment to evaluate BEAGLE in this extreme scenario. We suppose the subject model is injected with three backdoors, BadNets, Instagram filter, and WaNet. There are two cases for this multiple trojan scenario. In the first case, the three backdoors aim to attack three different classes. Then it is not much different from the model having only one backdoor, because it is easy to distinguish the trojaned samples of the three backdoors by checking their output labels. In the second case, the three backdoors aim to attack the same class. The problem is hence reduced to whether BEAGLE can decompose a trigger from a set of mixed trojaned samples. We have trained 10 trojaned models with 3 backdoors and 10 clean VGG-11 models on CIFAR-10. We assume BEAGLE has access to 3 additional trojaned models for decomposition and scanner synthesis using ABS. For each trojaned model, we use 10 trojaned samples with 4 BadNets, 3 Instagram filter and 3 WaNet. We leverage the patching function to decompose a trigger from these trojaned images. Note that the decomposed patch trigger achieves an average ASR of 91.7%, outperforming the decomposed transforming trigger with an ASR of 87.3%. The result shows that BEAGLE's synthesized scanner detects the backdoor well with 100% accuracy and no FP or FN. Although injecting multiple backdoors affects the decomposition process, it causes the trigger features to be more distinguishable and thus easy to detect by the scanner.

TABLE XVII: Ablation study of different number of given samples

Backdoor	N_P	N_C	Decomposed Clean Images				Decomposed Trigger			
			L1↑	PSNR↑	SSIM↑	ACC↑	L1↑	PSNR↑	SSIM↑	ASR↑
BadNets	2	100	0.052	24.11	0.95	1.0	0.032	24.89	0.90	0.99
	5	100	0.040	27.91	0.97	1.0	0.020	27.64	0.92	1.00
	10	100	0.032	31.74	0.97	1.0	0.018	28.17	0.94	1.00
	10	50	0.050	26.28	0.95	1.0	0.043	23.96	0.86	1.00
	10	20	0.061	27.71	0.95	1.0	0.116	20.15	0.75	1.00
	10	10	0.087	25.15	0.92	1.0	0.248	16.55	0.60	1.00
Dynamic	2	100	0.057	26.29	0.94	1.0	0.113	18.54	0.77	1.00
	5	100	0.060	25.06	0.94	1.0	0.104	18.39	0.79	1.00
	10	100	0.058	26.70	0.93	1.0	0.088	20.03	0.78	1.00
	10	50	0.066	24.97	0.92	1.0	0.103	19.34	0.74	1.00
	10	20	0.090	24.08	0.90	1.0	0.190	17.49	0.67	1.00
	10	10	0.107	23.06	0.88	1.0	0.297	15.63	0.55	1.00
Reflection	2	100	0.459	14.55	0.58	1.0	0.177	22.37	0.84	0.97
	5	100	0.342	17.48	0.77	1.0	0.199	22.00	0.87	0.96
	10	100	0.217	20.87	0.80	1.0	0.170	22.35	0.85	0.98
	10	50	0.225	20.60	0.80	1.0	0.217	20.63	0.78	0.94
	10	20	0.243	19.90	0.76	1.0	0.268	18.64	0.69	0.90
	10	10	0.255	19.44	0.74	1.0	0.287	18.61	0.69	0.85
WaNet	2	100	0.109	26.25	0.96	1.0	0.090	28.10	0.95	0.94
	5	100	0.119	25.83	0.94	1.0	0.113	26.69	0.94	0.91
	10	100	0.118	25.98	0.95	1.0	0.108	27.17	0.96	0.90
	10	50	0.120	25.84	0.94	1.0	0.106	27.27	0.96	0.78
	10	20	0.112	25.99	0.94	1.0	0.089	28.36	0.96	0.50
	10	10	0.131	24.94	0.93	1.0	0.111	26.75	0.96	0.28
Gotham	2	100	0.206	21.62	0.93	1.0	0.221	19.69	0.88	0.98
	5	100	0.187	22.27	0.91	1.0	0.199	20.10	0.89	0.99
	10	100	0.184	22.38	0.91	1.0	0.186	20.90	0.89	0.98
	10	50	0.189	22.15	0.90	1.0	0.196	20.37	0.90	0.96
	10	20	0.214	21.18	0.88	1.0	0.222	19.50	0.88	0.96
	10	10	0.222	20.68	0.87	1.0	0.207	20.14	0.88	0.95

TABLE XVIII: Scanning accuracy using different clustering methods

Method	K-means-S	K-means-E	GMM-S	DBSCAN
Number of Clusters	5	6	4	4
Accuracy	0.94	0.93	0.94	0.94

E. Ablation Study

Effect of Normalization in Attack Decomposition. Normalization helps ensure the values of decomposed clean images are within the distribution of clean validation images as defined in Eq. 7. We conduct the experiments on CIFAR-10 with VGG-11. Similar to Table XIII, we report the visual quality using L^1 distance, PSNR, SSIM, and classification accuracy using ACC, ASR for decomposed clean images and triggers. The top half of Table XIX shows the results. For each evaluation metric, we show the results without the normalization on the left and the results with normalization on the right. Observe that for most cases, using normalization can effectively improve the performance of the decomposition.

Comparison with Trigger Inversion Methods. In this study, we compare the quality of the decomposed triggers by BEAGLE with those directly inverted from trojaned models using existing trigger inversion techniques. The comparison is not to claim our results are better as BEAGLE leverages attack samples, but rather to provide a reference. Specifically, we compare with NC on patching attacks and ABS-filter on transforming attacks. The bottom half of Table XIX presents the results. Observe that BEAGLE outperforms NC on visual quality for patching backdoors and ABS-filter for transforming backdoors, especially for complex triggers such as Invisible, WaNet, and DFST. The decomposed triggers by BEAGLE can achieve >90% ASR, while those by ABS-filter can only achieve 20-70% ASR. This demonstrates that BEAGLE can effectively approximate the trigger injection function comparing to existing works.

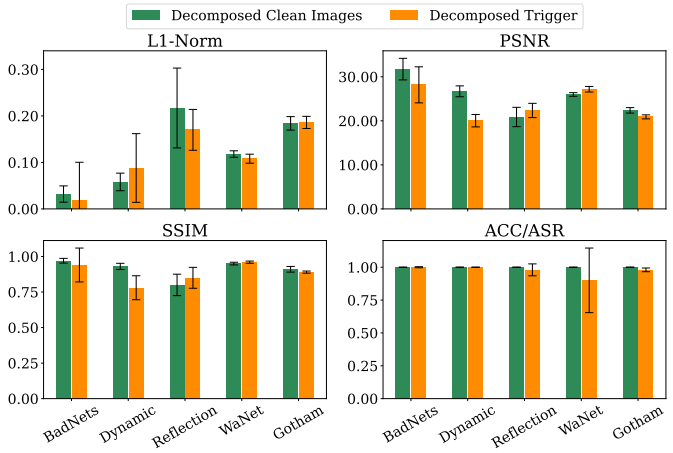


Fig. 22: Effect of different numbers of available inputs

Effect of Different Numbers of Available Inputs. We study the effect of using different numbers of trojaned and clean samples. We evaluate on using 2, 5, 10 trojaned images, and 10, 20, 50, 100 clean images. The other settings are kept the same and five attacks are used in the experiments. Figure 22 shows the results (details can be found in Table XVII). The x-axis in each subfigure denotes different attacks and the y-axis the value of the corresponding evaluation metric. The green bars are for the decomposed clean images and the orange bars for the decomposed triggers. The bars present the results for the original setting (10 trojaned images + 100 clean images), and the whiskers denote the results of using other settings. Observe that the variance is normally small in most cases, except for Reflection and WaNet. As BEAGLE leverages normalization for attack decomposition, the number of images may affect the approximation of trojaned and clean distributions and hence the final decomposition. The experiments show the number of available inputs has minor impact on BEAGLE.

Effect of Different Choices of Percentile of Distribution. We study the effect of different choices of percentile of distribution in the regularization term in Eq. 17 of the synthesized scanner. Take TrojAI round 3 as an example. We randomly select 100 clean models and 100 models attacked by polygon triggers, where the random seed is 1024. We follow the experiment setup presented in Section IV-B1 and use ABS as BEAGLE's downstream scanner. Besides the default choice of percentile range 15%-85%, we conduct two more experiments with percentile range 2%-98% and 30%-70%. Table XX shows the results. Observe that in all cases, BEAGLE can achieve high accuracy over 93%, and the choice of percentile is a trade-off between low false negative rate and low false positive rate. When the percentile is set at a large extent, e.g., 2%-98%, BEAGLE has a slightly high false positive rate 10%, and a low false negative rate 5%. On the other hand, when the percentile is constrained in a small extent, e.g., 30%-70%, the

TABLE XIX: Ablation study on normalization and trigger inversion

Technique	Backdoor	Decomposed Clean Images								Decomposed Trigger							
		L1 ↓		PSNR ↑		SSIM ↑		ACC ↑		L1 ↓		PSNR ↑		SSIM ↑		ASR ↑	
		Baseline	BG.	Baseline	BG.	Baseline	BG.	Baseline	BG.	Baseline	BG.	Baseline	BG.	Baseline	BG.	Baseline	BG.
Normalization	Reflection	0.662	**0.212**	11.97	**21.08**	0.62	**0.81**	1.0	**1.0**	0.425	**0.164**	15.31	**22.92**	0.70	**0.89**	0.92	**0.98**
	SIG	**0.142**	0.191	**24.25**	21.89	**0.86**	0.84	1.0	**1.0**	**0.063**	0.077	**29.62**	27.73	0.96	**0.96**	0.86	**0.94**
	Blend	0.304	**0.196**	18.47	**21.92**	0.81	**0.84**	1.0	**1.0**	0.214	**0.118**	21.61	**26.08**	0.88	**0.94**	0.99	**1.00**
	Invisible	**0.069**	0.091	**30.32**	28.29	0.96	**0.96**	1.0	**1.0**	**0.071**	0.099	**29.39**	27.12	**0.93**	0.88	0.74	**0.92**
	WaNet	**0.067**	0.116	**29.25**	26.20	**0.96**	0.95	1.0	**1.0**	**0.054**	0.101	**31.49**	27.68	0.97	**0.97**	0.89	**0.90**
	Gotham	0.414	**0.183**	15.68	**22.49**	0.81	**0.91**	1.0	**1.0**	0.432	**0.185**	15.05	**20.96**	0.75	**0.90**	0.82	**0.97**
	DFST	0.593	**0.454**	13.52	**15.83**	0.65	**0.74**	1.0	**1.0**	0.734	**0.383**	11.94	**16.41**	0.61	**0.68**	0.92	**0.92**
NC	BadNets	0.040	**0.030**	28.33	**31.14**	0.97	**0.97**	1.0	**1.0**	0.082	**0.016**	17.84	**28.87**	0.78	**0.96**	1.00	**1.00**
	TrojNN	0.031	**0.025**	29.84	**33.41**	0.97	**0.98**	1.0	**1.0**	0.109	**0.008**	15.93	**30.58**	0.58	**0.98**	1.00	**1.00**
	Dynamic	0.055	**0.054**	**27.29**	25.83	0.94	**0.94**	1.0	**1.0**	0.105	**0.081**	18.15	**20.18**	0.73	**0.81**	1.00	**1.00**
ABS-filter	Invisible	0.104	**0.091**	27.00	**28.29**	0.95	**0.96**	1.0	**1.0**	0.366	**0.099**	15.41	**27.12**	0.55	**0.88**	0.41	**0.92**
	WaNet	0.122	**0.116**	25.74	**26.20**	0.95	**0.95**	1.0	**1.0**	0.207	**0.101**	21.83	**27.68**	0.84	**0.97**	0.21	**0.90**
	Gotham	**0.177**	0.183	**22.69**	22.49	0.91	**0.91**	1.0	**1.0**	0.195	**0.185**	20.86	**20.96**	0.89	**0.90**	0.97	**0.97**
	DFST	**0.412**	0.454	**16.47**	15.83	**0.75**	0.74	1.0	**1.0**	0.549	**0.383**	13.72	**16.41**	0.59	**0.68**	0.68	**0.92**

TABLE XX: Ablation study of different choices of distribution percentile

Percentile	TP	FN	TN	FP	ACC
2%-98%	95	5	90	10	0.93
15%-85%	91	9	96	4	0.94
30%-70%	91	9	96	4	0.94

TABLE XXI: Ablation study of using different pre-trained GANs

Dataset	Decomposed Clean Images				Decomposed Trigger			
	L1 ↓	PSNR ↑	SSIM ↑	ACC ↑	L1 ↓	PSNR ↑	SSIM ↑	ASR ↑
CIFAR-10	0.212	21.08	0.81	1.0	0.164	22.92	0.89	0.98
GTSRB	0.321	17.84	0.58	1.0	0.294	18.13	0.76	0.96
CelebA	0.373	15.66	0.42	1.0	0.263	16.64	0.63	0.97

false positive rate is reduced to 4% while the false negative rate increases to 9%. This is reasonable as a large percentile range poses less regularization penalty on trigger inversion and thus more corner case triggers can be inverted. However, a small percentile range penalizes more on trigger inversion and hence fewer natural trojans (false positive cases) are induced. The trade-off here is that whether the user prefers a low false positive rate or a low false negative rate. Typically, we set the default percentile as 15%-85% for a balanced performance.

Effect of Using Different Pre-trained GANs in Attack Decomposition. In this study, we aim to show that BEAGLE's decomposition process may not require that the GAN is well-trained on the model input domain. We conduct an experiment in which we leverage a pre-trained GAN on CIFAR-10 to decompose 10 Reflection attack instances whose source images come from CIFAR-10, GTSRB and CelebA. Note that GTSRB and CelebA have no overlapping classes with CIFAR-10. Table XXI shows the results. The first column denotes the dataset and the following columns measure the decomposition performance, same as Table XIII. The first row shows the result when the decomposition process leverages the GAN pre-trained on CIFAR-10, the model input domain. Observe that the decomposition performance is very good with a small L^1 error, high PSNR, SSIM scores and high accuracy/ASR for both decomposed clean images and triggers. The second row illustrates the decomposition performance of GTSRB instances using the same GAN. We can see that there is a medium degradation on the decomposed clean images, with -0.23 SSIM score difference, and a small degradation on the decomposed trigger, with -0.13 SSIM score difference. In a more extreme case that we use the GAN trained on CIFAR-10 with an input size 32×32 to generate high-resolution images from CelebA with an input size 128×128. The decomposition performance is further reduced, with -0.39 SSIM score difference on the decomposed clean images and -0.26 on the decomposed trigger, compared to the results on CIFAR-10 instances. However, in both cases, the decomposed clean images can achieve 100% accuracy and the decomposed triggers can achieve > 95% ASR. We argue that the degradation on visual quality of decomposed clean images is acceptable as finally the decomposed trigger is used for scanner synthesis. Besides, since the decomposed triggers achieve a high ASR, they are sufficient for further attack summarization.

TABLE XXII: Ablation study of different choices of α

α	Decomposed Clean Images				Decomposed Trigger			
	L1 ↓	PSNR ↑	SSIM ↑	ACC ↑	L1 ↓	PSNR ↑	SSIM ↑	ASR ↑
0.1	0.702	12.78	0.39	0.7	0.650	11.99	0.32	0.99
10^2	0.217	20.84	0.80	1.0	0.186	21.83	0.85	0.98
10^5	0.221	20.75	0.80	0.8	0.175	22.95	0.88	0.82

Effect of Different Choices of Parameter α in Attack Decomposition. Parameter α in Eq. 4 controls the trade-off between the reconstruction quality and the trigger effectiveness (Accuracy/ASR) during attack decomposition. We take the decomposition of Reflection backdoor on CIFAR-10 as an example to study the effect of setting different values of α. Table XXII shows the results. The first column denotes the choice of α and the following columns show the decomposition performance. Observe that when α is small, e.g., 0.1, the reconstruction quality is low, achieving near 0.3 SSIM score while the ASR of decomposed trigger is high, reaching nearly 100%. On the other hand, when α is large, e.g., 10^5, the reconstruction quality tends to be high, increasing to an SSIM score of 0.8. However, the trigger effectiveness is reduced, i.e., 0.16 ASR degradation. Overall, we find that $\alpha = 10^2$ provides

the best trade-offs between the visual reconstruction quality and accuracy/ASR. Hence, we set $\alpha = 10^2$ for Eq. 4.