CT-GAN: Malicious Tampering of 3D Medical Imagery using Deep Learning

Yisroel Mirsky and Tom Mahler, ***Ben-Gurion University;*** **Ilan Shelef,**
Soroka University Medical Center; **Yuval Elovici,** ***Ben-Gurion University***

https://www.usenix.org/conference/usenixsecurity19/presentation/mirsky

This paper is included in the Proceedings of the 28th USENIX Security Symposium.

August 14–16, 2019 • Santa Clara, CA, USA

978-1-939133-06-9

Open access to the Proceedings of the 28th USENIX Security Symposium is sponsored by USENIX.

CT-GAN: Malicious Tampering of 3D Medical Imagery using Deep Learning

Yisroel Mirsky[1], Tom Mahler[1], Ilan Shelef[2], and Yuval Elovici[1]
[1]Department of Information Systems Engineering, Ben-Gurion University, Israel
[2]Soroka University Medical Center, Beer-Sheva, Israel
yisroel@post.bgu.ac.il, mahlert@post.bgu.ac.il, shelef@bgu.ac.il, and elovici@bgu.ac.il

Abstract

In 2018, clinics and hospitals were hit with numerous attacks leading to significant data breaches and interruptions in medical services. An attacker with access to medical records can do much more than hold the data for ransom or sell it on the black market.

In this paper, we show how an attacker can use deep-learning to add or remove evidence of medical conditions from volumetric (3D) medical scans. An attacker may perform this act in order to stop a political candidate, sabotage research, commit insurance fraud, perform an act of terrorism, or even commit murder. We implement the attack using a 3D conditional GAN and show how the framework (CT-GAN) can be automated. Although the body is complex and 3D medical scans are very large, CT-GAN achieves realistic results which can be executed in milliseconds.

To evaluate the attack, we focused on injecting and removing lung cancer from CT scans. We show how three expert radiologists and a state-of-the-art deep learning AI are highly susceptible to the attack. We also explore the attack surface of a modern radiology network and demonstrate one attack vector: we intercepted and manipulated CT scans in an active hospital network with a covert penetration test.

1 Introduction

Medical imaging is the non-invasive process of producing internal visuals of a body for the purpose of medical examination, analysis, and treatment. In some cases, volumetric (3D) scans are required to diagnose certain conditions. The two most common techniques for producing detailed 3D medical imagery are Magnetic Resonance Imaging (MRI), and CT (Computed Tomography). Both MRI and CT scanner are essential tools in the medical domain. In 2016, there were approximately 38 million MRI scans and 79 million CT scans performed in the United States [1].[1]

MRI and CT scanners are similar in that they both create 3D images by taking many 2D scans of the body over the axial plane (from front to back) along the body. The difference between the two is that MRIs use powerful magnetic fields and CTs use X-Rays. As a result, the two modalities capture body tissues differently: MRIs are used to diagnose issues with bone, joint, ligament, cartilage, and herniated discs. CTs are used to diagnose cancer, heart disease, appendicitis, musculoskeletal disorders, trauma, and infectious diseases [2].

Today, CT and MRI scanners are managed though a picture archiving and communication system (PACS). A PACS is essentially an Ethernet-based network involving a central server which (1) receives scans from connected imaging devices, (2) stores the scans in a database for later retrieval, and (3) retrieves the scans for radiologists to analyze and annotate. The digital medical scans are sent and stored using the standardized DICOM format.[2]

1.1 The Vulnerability

The security of health-care systems has been lagging behind modern standards [3–6]. This is partially because health-care security policies mostly address data privacy (access-control) but not data security (availability/integrity) [7]. Some PACS are intentionally or accidentally exposed to the Internet via web access solutions. Some example products include Centricity PACS (GE Healthcare), IntelliSpace (Philips), Synapse Mobility (FujiFilm), and PowerServer (RamSoft). A quick search on Shodan.io reveals 1,849 medical image (DICOM) servers and 842 PACS servers exposed to the Internet. Recently, a researcher at McAfee demonstrated how these web portals can be exploited to view and modify a patient's 3D DICOM imagery [8]. PACS which are not directly connected to the Internet are indirectly connected via the facility's internal network [9]. They are also vulnerable to social engineering attacks, physical access, and insiders [10].

Therefore, a motivated attacker will likely be able to access a target PACS and the medical imagery within it. Later in section 4 we will discuss the attack vectors in greater detail.

1.2 The Threat

An attacker with access to medical imagery can alter the contents to cause a misdiagnosis. Concretely, the attacker can

[1]245 CT scans and 118 MRI scans per 1,000 inhabitants.

[2]https://www.dicomstandard.org/about/

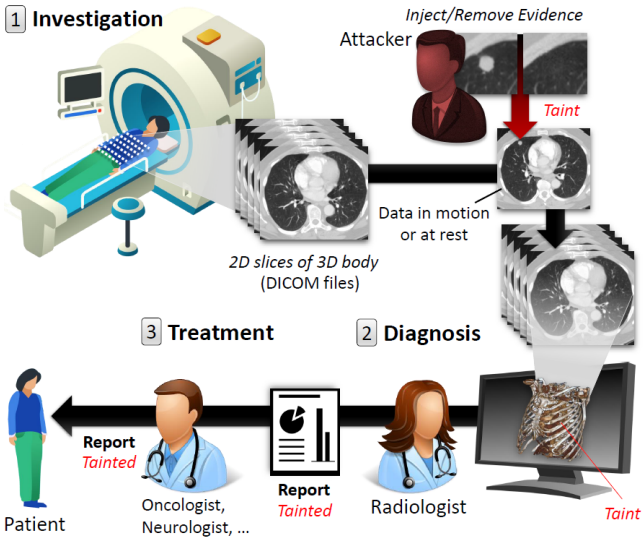

Figure 1: By tampering with the medical imagery between the investigation and diagnosis stages, both the radiologist and the reporting physician believe the fallacy set by the attacker.

add or remove evidence of some medical condition. Fig. 1 illustrates this process where an attacker injects/removes lung cancer from a scan.

Volumetric medical scans provide strong evidence of medical conditions. In many cases, a patient may be treated based on this evidence without the need to consider other medical tests. For example, some lesions are obvious or require immediate surgery. Moreover, some lesions will legitimately not show up on other medical tests (e.g., meniscus trauma and some breast cancers). Regardless, even if other tests aren't usually negative, ultimately, the evidence in the scan will be used to diagnose and treat the patient. As a result, an attacker with access to a scan has the power to change the outcome of the patient's diagnosis. For example, an attacker can add or remove evidence of aneurysms, heart disease, blood clots, infections, arthritis, cartilage problems, torn ligaments or tendons, tumors in the brain, heart, or spine, and other cancers.

There are many reasons why an attacker would want to alter medical imagery. Consider the following scenario: An individual or state adversary wants to affect the outcome of an election. To do so, the attacker adds cancer to a CT scan performed on a political candidate (the appointment/referral can be pre-existing, setup via social engineering, or part of a lung cancer screening program). After learning of the cancer, the candidate steps-down from his or her position. The same scenario can be applied to existing leadership.

Another scenario to consider is that of ransomware: An attacker seeks out monetary gain by holding the integrity of the medical imagery hostage. The attacker achieves this by altering a few scans and then by demanding payment for revealing which scans have been affected.

Furthermore, consider the case of insurance fraud: Somebody alters his or her own medical records in order to receive money directly from his or her insurance company, or receive

Table 1: Summary of an attacker's motivations and goals for injecting/removing evidence in 3D medical imagery.

+ : Add Evidence
− : Remove Evidence
± : Either

● : Target Effect
○ : Side Effect

			Goal: Steal Job Position	Affect Elections	Remove Leader	Sabotage Research	Falsify Research	Hold Data Hostage	Insurance Fraud	Murder	Terrorize
Motivation		Ideological			+						±
		Political		+	+					−	
		Money	+		+	+	−	±	+		
		Fame/Attn.	+		+		±				
		Revenge	+		+					−	+
Effect	Physical	Injury	○	○	○		○	○		○	●
		Death						○		●	●
	Mental	Trauma	○	○	○			○		○	●
		Life Course	●	●	●	○			○	○	●
	Monetary	Cause Loss	○		○	●		○		○	●
		Payout	●			○	●	●	●		
Attack Type		Untargeted				X		X			X
		Targeted	X	X	X	X	X		X	X	

handicap benefits (e.g., lower taxes etc.) In this case, there is no risk of physical injury to others, and the payout can be very large. For example, one can (1) sign up for disability/life insurance, then (2) fake a car accident or other incident, (3) complain of an inability to work, sense, or sleep, then (4) add a small brain hemorrhage or spinal fracture to his or her own scan during an investigation (this evidence is very hard to refute), and then (5) file a claim and receive cash from the insurance company.[3]

There are many more reasons why an attacker would want to tamper with the imagery. For example: falsifying research evidence, sabotaging another company's research, job theft, terrorism, assassination, and even murder.

Depending on the attacker's goal, the attack may be either untargeted or targeted:

Untargeted Attacks are where there is no specific target patient. In this case, the attacker targets a victim who is receiving a random voluntary cancer screening, is having an annual scan (e.g., BRACA patients, smokers...), or is being scanned due to an injury. These victims will either have an 'incidental finding' when the radiologist reviews the scan (injection) or are indeed sick but the evidence won't show (removal).

Targeted Attacks are where there is a specific target patient. In these attacks, the patient may be lured to the hospital for a scan. This can be accomplished by (1) adding an appointment in the system, (2) crafting a cancer screening invite, (3) spoofing the patient's doctor, or (4) tampering/appending the patient's routine lab tests. For

[3] For example, see products such as AIG's Quality of Life insurance.

example, high-PSA in blood indicates prostate cancer leading to an **abdominal MRI**, high thyrotropin in blood indicates a brain tumor leading to a **head MRI**, and metanephrine in urine of hypertensive patients indicates cancer/tumor leading to a **chest/abdominal CT**

In this paper we will focus on the injection and removal of lung cancer from CT scans. Table 1 summarizes attacker's motivations, goals, and effects by doing so. The reason we investigate this attack is because lung cancer is common and has the highest mortality rate [11]. Therefore, due its impact, an attacker is likely to manipulate lung cancer to achieve his or her goal. We note that the threat, attack, and countermeasures proposed in this paper also apply to MRIs and medical conditions other than those listed above.

1.3 The Attack

With the help of machine learning, the domain of image generation has advanced significantly over the last ten years [12]. In 2014, there was a breakthrough in the domain when Goodfellow et al. [13] introduced a special kind of deep neural network called a generative adversarial network (GAN). GANs consist of two neural networks which work against each other: the *generator* and the *discriminator*. The *generator* creates fake samples with the aim of fooling the *discriminator*, and the *discriminator* learns to differentiate between real and fake samples. When applied to images, the result of this game helps the *generator* create fake imagery which are photo realistic. While GANs have been used for positive tasks, researchers have also shown how they can be used for malicious tasks such as malware obfuscation [14, 15] and misinformation (e.g., deepfakes [16]).

In this paper, we show how an attacker can realistically inject and remove medical conditions with 3D CT scans. The framework, called CT-GAN, uses two conditional GANs (cGAN) to perform in-painting (image completion) [17] on 3D imagery. For injection, a cGAN is trained on unhealthy samples so that the *generator* will always complete the images accordingly. Conversely, for removal, another cGAN is trained on healthy samples only.

To make the process efficient and the output anatomically realistic, we perform the following steps: (1) locate where the evidence should be inject/removed, (2) cut out a rectangular cuboid from the location, (3) interpolate (scale) the cuboid, (4) modify the cuboid with the cGAN, (5) rescale, and (6) paste it back into the original scan. By dealing with a small portion of the scan, the problem complexity is reduced by focusing the GAN on the relevant area of the body (as opposed to the entire CT). Moreover, the algorithm complexity is reduced by processing fewer inputs[4] (pixels) and concepts (anatomical features). This results in fast execution and high anatomical realism. The interpolation step is necessary because the scale of a scan can be different between patients. To compensate for the resulting interpolation blur, we mask the relevant content according to water density in the tissue (Hounsfield units) and hide the smoothness by adding Gaussian white noise. In order to assist the GAN in generating realistic features, histogram equalization is performed on the input samples. We found that this transformation helps the 3D convolutional neural networks in the GAN learn how to generate the subtle features found in the human body. The entire process is automated, meanings that the attack can be deployed in an air gapped PACS.

To verify the threat of this attack, we trained CT-GAN to inject/remove lung cancer and hired three radiologists to diagnose a mix of 70 tampered and 30 authentic CT scans. The radiologists diagnosed 99% of the injected patients with malign cancer, and 94% of cancer removed patients as being healthy. After informing the radiologists of the attack, they still misdiagnosed 60% of those with injections, and 87% of those with removals. In addition to the radiologists, we also showed how CT-GAN is an effective adversarial machine learning attack. We found that the state-of-the-art lung cancer screening model misdiagnosed 100% of the tampered patients. Thus, cancer screening tools, used by some radiologists, are also vulnerable to this attack.

This attack is a concern because infiltration of healthcare networks has become common [3], and internal network security is often poor [18]. Moreover, for injection, the attacker is still likely to succeed even if medical treatment is not performed. This is because many goals rely on simply scaring the patient enough to affect his/her daily/professional life. For example, even if an immediate deletion surgery is not deemed necessary based on the scan and lab results, there will still be weekly/monthly follow-up scans to track the tumor's growth. This will affect the patient's life given the uncertainty of his or her future.

1.4 The Contribution

To the best of our knowledge, it has not been shown how an attacker can maliciously alter the content of a 3D medical image in a realistic and automated way. Therefore, this is the first comprehensive research which exposes, demonstrates, and verifies the threat of an attacker manipulating 3D medical imagery. In summary, the contributions of this paper are as follows:

The Attack Model We are the first to present how an attacker can infiltrate a PACS network and then use malware to autonomously tamper 3D medical imagery. We also provide a systematic overview of the attack, vulnerabilities, attack vectors, motivations, and attack goals. Finally, we demonstrate one possible attack vector through a penetration test performed on a hospital where we covertly connect a man-in-the-middle device to an actual CT scanner. By performing this pen-test, we provide insights into the security of a modern hospital's internal network.

Attack Implementation We are the first to demonstrate how GANs, with the proper preprocessing, can be used to efficiently, realistically, and automatically inject/remove lung cancer into/from large 3D CT scans. We also evaluate how well the algorithm can deceive both humans and

[4] A 3D CT scan can have over 157 million pixels whereas the latest advances in GANs can only handle about 2 million pixels (HD images).

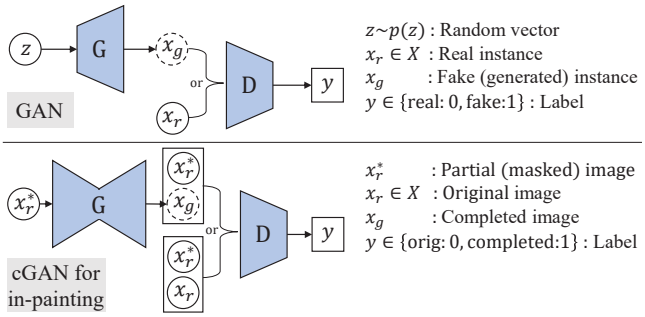

Figure 2: A schematic view of a classic GAN (top) and a cGAN setup for in-painting.

machines: radiologists and state-of-the-art AI. We also show how this implementation might be used by an attacker since it can be automated (in the case of an air gapped system) and is fast (in the case of an infected DICOM viewer).

Countermeasures We enumerate various countermeasures which can be used to mitigate the threat. We also provide the reader with best practices and configurations which can be implemented immediately to help prevent this attack.

For reproducibility and further investigation, we have published our tampered datasets and source code online[5] along with a pen-test video.[6]

The remainder of the paper is organized as follows: First we present a short background on GANs. Then, in section 3, we review related works and contrast them ours. In section 4 we present the attack model and demonstrate one of the attack vectors. In section 5, we present CT-GAN's neural architecture, its attack process, and some samples. In section 6 we evaluate the quality of the manipulations and asses the threat of the attack. Finally, in sections 7 and 8 we present countermeasures and our conclusion.

2 Background: GANs

The most basic GAN consists of two neural networks: the *generator* (G) and *discriminator* (D). The objective of the GAN is to generate new images which are visually similar to real images found in a sample data distribution X (i.e., a set of images). The input to G is the random noise vector z drawn from the prior distribution $p(z)$ (e.g., a Gaussian distribution). The output of G, denoted x_g, is an image which is expected to have visual similarity with those in X. Let the non-linear function learned by G parametrized by θ_g be denoted as $x_g = G(z;\theta_g)$. The input to D is either a real image $x_r \in X$ or a generated image $x_g \in G(Z;\theta_g)$. The output of D is the probability that x_g is real or fake. Let the non-linear function learned by D parametrized by θ_d be denoted as $x_d = D(x;\theta_d)$. The top of Fig. 2 illustrates the configuration of a classic GAN.

It can be seen that D and G are playing a zero-sum game where G is trying to find better (more realistic) samples to fool D, while D is learning to catch every fake sample generated by G. After training, D is discarded and G is used to generate new samples.

A cGAN is a GAN which has a *generator* and *discriminator* conditioned on an additional input (e.g., class labels). This input extends the latent space z with further information thus assisting the network to generate and discriminate images better. In [17] the authors propose an image-to-image translation framework using cGANs (a.k.a. pix2pix). There the authors showed how deep convolutional cGANs can be used to translate images from one domain to another. For example, converting casual photos to a Van Gogh paintings.

One application of the pix2pix framework is in-painting; the process of completing a missing part of an image. When using pix2pix to perform in-painting, the *generator* tries to fill in a missing part of an image based on the surrounding context, and its past experience (other images seen during training). Meanwhile, the *discriminator* tries to differentiate between completed images and original images, given the surrounding context. Concretely, the input to G is a copy of x_r where missing regions of the image are replaced with zeros. We denote this masked input as x_r^*. The output of G is the completed image, visually similar to those in X. The input to D is either the concatenation (x_r^*, x_r) or $(x_r^*, G(x_r^*;\theta_g))$. The bottom of Fig. 2 illustrates the described cGAN. The process for training this kind of GAN is as follows:

Training Procedure for cGAN In-painting

Repeat for k training iterations:

1. Pull a random batch of samples $x_r \in X$, and mask the samples with zeros to produce the respective x_r^*.
2. **Train** D:
   - 2.1. Forward propagate (x_r^*, x_r) through D, compute the error given the label $y = 0$, and back propagate the error through D to update θ_d.
   - 2.2. Forward propagate $(x_r^*, G(x_r^*;\theta_g))$ through D, compute the error given the label $y = 1$, and back propagate the error through D to update θ_d.
3. **Train** G:
   - 3.1. Forward propagate x_r^* through G and then $(x_r^*, G(x_r^*;\theta_g))$ through D, compute the error at the output of D given the label $y = 0$, back propagate the error through D to G without updating θ_d, and continue the back propagation through G while updating θ_g.

Although pix2pix does not use a latent random input z, it avoids deterministic outputs by performing random dropouts in the generator during training. this forces the network to learn multiple representations of the data.

We note that there is a GAN called a CycleGAN [19] that can directly translate images between two domains (e.g., benign $\leftrightarrow$ malign). However, we found that the CycleGAN was unable to inject realistic cancer into 3D samples. Therefore, we opted

[5] https://github.com/ymirsky/CT-GAN
[6] https://youtu.be/_mkRAArj-x0

to use the pix2pix model for in-painting because it produced much better results. This may be due to the complexity of the anatomy in the 3D samples and the fact that we had relatively few training samples. Since labeled datasets contain at most a few hundred scans, our approach is more likely to be used by an attacker. Another reason is that in-painting is arguably easier to perform than 'style transfer' when considering different bodies. Regardless, in-painting ensures that the final image can be seamlessly pasted back into the scan without border effects.

3 Related Work

The concept of tampering medical imagery, and the use of GANs on medical imagery, is not new. In this section we briefly review these subjects and compare prior results to our work.

3.1 Tampering with Medical Images

Many works have proposed methods for detecting forgeries in medical images [20], but none have focused on the attack itself. The most common methods of image forgery are: copying content from one image to another (image splicing), duplicating content within the same image to cover up or add something (copy-move), and enhancing an image to give it a different feel (image retouching) [21].

Copy-move attacks can be used to cover up evidence or duplicate existing evidence (e.g., a tumor). However, duplicating evidence will raise suspicion because radiologists closely analyze each discovered instance. Image-splicing can be used to copy evidence from one scan to another. However, CT scanners have distinct local noise patterns which are visually noticeable [22, 23]. The copied patterns would not fit the local pattern and thus raise suspicion. More importantly, both copy-move and image-splicing techniques are performed using 2D image editing software such as Photoshop. These tools require a digital artist to manually edit the scan. Even if the attacker has a digital artist, it is hard to accurately inject and remove cancer realistically. This is because human bodies are complex and diverse. For example, cancers and tumors are usually attached to nearby anatomy (lung walls, bronchi, etc.) which may be hard to alter accurately under the scrutiny of expert radiologists. Another consideration is that CT scans are 3D and not 2D, which adds to the difficulty. It is also important to note that an attacker will likely need to automate the entire process in a malware since (1) many PACS are not directly connected to the Internet and (2) the diagnosis may occur immediately after the scan is performed.

In contrast to the Photoshopping approach, CT-GAN (1) works on 3D medical imagery, which provide stronger evidence than a 2D scans, (2) realistically alters the contents of a 3D scan while considering nearby anatomy, and (3) can be completely automated. The latter point is important because (1) some PACS are not directly connected to the Internet, (2) diagnosis can happen right after the actual scan, (3) the malware may be inside the radiologist's viewing app.

3.2 GANs in Medical Imagery

Since 2016, over 100 papers relating to GANs and medical imaging have been published [24]. These publications mostly relate image reconstruction, denoising, image generation (synthesis), segmentation, detection, classification, and registration. We will focus on the use of GANs to generate medical images.

Due to privacy laws, it is hard to acquire medical scans for training models and students. As a result, the main focus of GANs in this domain has been towards augmenting (expanding) datasets. One approach is to convert imagery from one modality to another. For example, in [25] the authors used cGANs to convert 2D slices of CT images to Positron Emission Tomography (PET) images. In [26, 27] the authors demonstrated a similar concept using a fully convolutional network with a cGAN architecture. In [28], the authors converted MRI images to CT images using domain adaptation. In [29], the authors converted MRI to CT images and vice versa using a CycleGAN.

Another approach to augmenting medical datasets is the generation of new instances. In [30], the authors use a deep convolutional GAN (DCGAN) to generate 2D brain MRI images with a resolution of 220x172. In [31], the authors used a DCGAN to generate 2D liver lesions with a resolution of 64x64. In [32], the authors generated 3D blood vessels using a Wasserstien (WGAN). In [33], the authors use a Laplace GAN (LAPGAN) to generate skin lesion images with 256x256 resolution. In [34], the authors train two DCGANs for generating 2D chest X-rays (one for malign and the other for benign). However, in [34], the generated samples were down sampled to 128x128 in resolution since this approach could not be scaled to the original resolution of 2000x3000. In [35] the authors generated 2D images of pulmonary lung nodules (lung cancer) with 56x56 resolution. The author's motivation was to create realistic datasets for doctors to practice on. The samples were generated using a DCGAN and their realism was assessed with help of two radiologists. The authors found that the radiologists were unable to accurately differentiate between real and fake samples.

These works contrast to our work in the following ways:

1. We are the first to introduce the use of GANs as a way to tamper with 3D imagery. The other works focused on synthesizing cancer samples for boosting classifiers, experiments, and training students, but not for malicious attacks. We also provide an overview of how the attack can be accomplished in a modern medical system.

2. All of the above works either generate small regions of a scan without the context of a surrounding body or generate a full 2D scan with a very low resolution. Samples which are generated without a context cannot be realistically 'pasted' back into any arbitrary medical scan. We generate/remove content realistically within existing bodies. Moreover, very low-resolution images of full scans cannot replace existing ones without raising suspicion (especially if the body doesn't match the actual person).

Our approach can modify full resolution 3D scans,[7] and the approach can be easily extended to 2D as well.

3. We are the first to evaluate how well a GAN can fool expert radiologists and state-of-the-art AI in full 3D lung cancer screening. Moreover, in our evaluation, the radiologists and AI were able to consider how the cancer was attached and placed within the surrounding anatomy.

4 The Attack Model

In this section we explore the attack surface by first presenting the network topology and then by discussing the possible vulnerabilities and attack vectors. We also demonstrate one of the attack vectors on an actual CT scanner.

4.1 Network Topology

In order to discuss the attack vectors we must first present the PACS network topology. Fig. 3 presents the common network configuration of PACS used in hospitals. The topology is based on PACS literature [9, 36–38], PACS enterprise solutions (e.g., Carestream), and our own surveys conducted on various hospitals. We note that, private medical clinics may have a much simpler topology and are sometimes directly connected to the Internet [8].

The basic elements of a PACS are as follows:

PACS Server. The heart of the PACS system. It is responsible for storing, organizing, and retrieving DICOM imagery and reports commonly via SQL. Although the most facilities use local servers, a few hospitals have transitioned to cloud storage [39].

RIS Server. The radiology information system (RIS) is responsible for managing medical imagery and associated data. Its primary use is for tracking radiology imaging orders and the reports of the radiologists. Doctors in the hospital's internal network can interface with the RIS to order scans, receive the resulting reports, and to obtain the DICOM scans [40].

Modality Workstation. A PC (typically Windows) which is used to control an imaging modality such as a CT scanner. During an appointment, the attending technician configures and captures the imagery via the workstation. The workstation sends all imagery in DICOM format to the PACS server for storage.

Radiologist Workstation. A radiologist can retrieve and view scans stored on the PACS server from various locations. The most common location is a viewing workstation within the department. Other locations include the radiologist's personal PC (local or remote via VPN), and sometimes a mobile device (via the Internet or within the local network).

[7] A CT scan can have a resolution from 512x512x600 to 1024x1024x600 and larger.

Web Server. An optional feature which enables radiologists to view of DICOM scans (in the PACS server) over the Internet. The content may be viewed though a web browser (e.g., medDream and Orthanc [41]), an app on a mobile device (e.g., FujiFilm's Synapse Mobility), or accessed via a web API (e.g., Dicoogle [42]).

Administrative Assistant's PC. This workstation has both Internet access (e.g., for emails) and access to the PACS network. Access to the PACS is enabled so that the assistant can maintain the devices' schedules: When a patient arrives at the imaging modality, for safety reasons, the technician confirms the patient's identity with the details sent to the modality's workstation (entered by the assistant). This ensures that the scans are not accidentally mixed up between the patients.

Hospital Network. Other departments within the hospital usually have access to the PACS network. For example, Oncology, Cardiology, Pathology, and OR/Surgery. In these cases, various workstations around the hospital can load DICOM files from the server given the right credentials. Furthermore, it is common for a hospital to deploy Wi-Fi access points, which are connected to the internal network, for employee access.

4.2 Attack Scenario

The attack scenario is as follows: An attacker wants to achieve one of the goals listed in Table 1 by injecting/removing medical evidence. In order to cause the target effect, the attacker will alter the contents of the target's CT scan(s) before the radiologist performs his or her diagnosis. The attacker will achieve this by either targeting the data-at-rest or data-in-motion.

Thedata-at-rest refers to the DICOM files stored on the PACS Server, or on the radiologist's personal computer (saved for later viewing). In some cases, DICOM files are stored on DVDs and then transferred to the hospital by the patient or an external doctor. Although the DVD may be swapped by the attacker, it is more likely the interaction will be via the network. The data-in-motion refers to DICOM files being transferred across the network or loaded into volatile memory by an application (e.g., a DICOM viewer).

We note that this scenario does not apply to the case where the goal is to falsify or sabotage research. Moreover, for insurance fraud, an attacker will have a much easier time targeting a small medical clinic. For simplicity, we will assume that the target PACS is in a hospital.

4.3 Target Assets

To capture/modify a medical scan, an attacker must compromise at least one of the assets numbered in Fig. 3. By compromising one of (1-4), the attacker gains access to every scan. By compromising (5) or (6), the attacker only gains access to a subset of scans. The RIS (3) can give the attacker full control over the PACS server (2), but only if the attacker can obtain the right

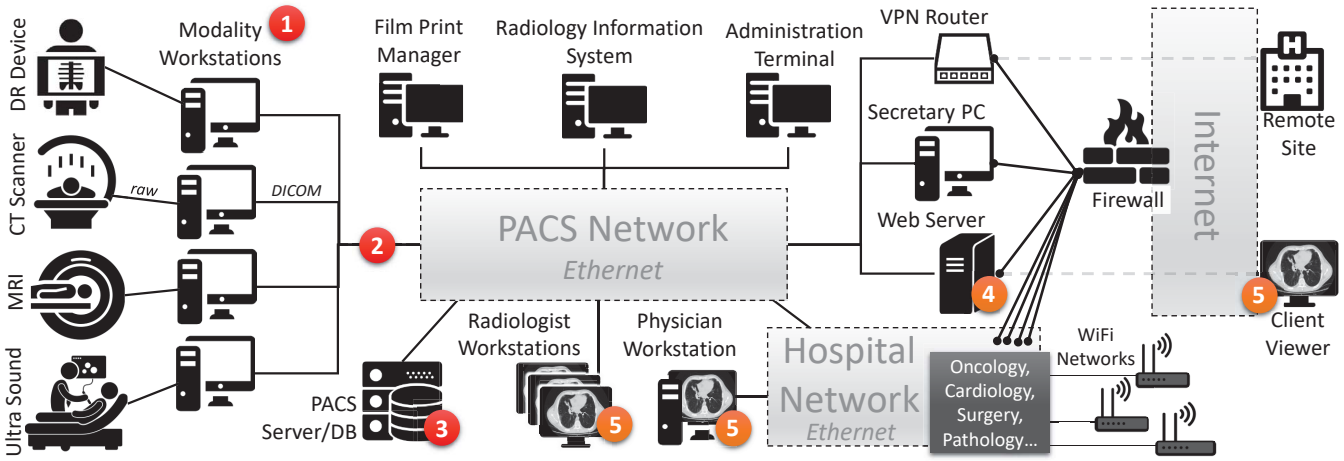

Figure 3: A network overview a PACS in a hospital. 1-3: points where an attacker can tamper with all scans. 4-5: points where an attacker can tamper with a subset of scans.

credentials or exploit the RIS software. The network wiring between the modalities and the PACS server (4) can be used to install a man-in-the-middle device. This device can modify data-in-motion if it is not encrypted (or if the protocol is flawed).

In all cases, it is likely that the attacker will infect the target asset with a custom malware, outlined in Fig. 4. This is because there may not be a direct route to the PACS via the Internet or because the diagnosis may take place immediately after the scan is taken.

4.4 Attack Vectors

There are many ways in which an attacker can reach the assets marked in Fig. 3. In general, the attack vectors involve either remote or local infiltration of the facility's network.

Remote Infiltration. The attacker may be able to exploit vulnerabilities in elements facing the Internet, providing the attacker with direct access to the PACS from the Internet (e.g., [8]). Another vector is to perform a social engineering attack. For example, a spear phishing attack on the department's administrative assistant to infect his/her workstation with a backdoor, or a phishing attack on the technician to have him install fraudulent updates.

If the PACS is not directly connected to the Internet, an alternative vector is to (1) infiltrate the hospital's internal network and then (2) perform lateral movement to the PACS. This is possible because PACS is usually connected to the internal network (using static routes and IPs), and the internal network is connected to the Internet (evident from the recent wave of

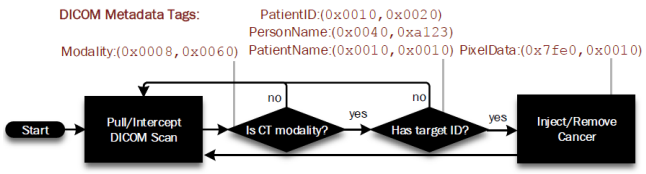

Figure 4: The tampering process of an autonomous malware.

cyber-attacks on medical facilities [3, 43–45]). The bridge between the internal network and the PACS is to enable doctors to view scans/reports and to enable the administrative assistant to manage patient referrals [9]. Another vector from the Internet is to compromise a remote site (e.g., a partnered hospital or clinic) which is linked to the hospital's internal network. Furthermore, the attacker may also try to infect a doctor's laptop or phone with a malware which will open a back door into the hospital.

If the attacker knows that radiologist analyzes scans on his or her personal computer, then the attacker can infect the radiologist's device or DICOM viewer remotely with the malware.

Local Infiltration. The attacker can gain physical access to the premises with a false pretext, such as being a technician from Philips who needs to run a diagnostic on the CT scanner. The attacker may also hire an insider or even be an insider. A recent report shows that 56% of cyber attacks on the healthcare industry come from internal threats [10].

Once inside, the attacker can plant the malware or a back door by (1) connecting a device to exposed network infrastructure (ports, wires, ...) [46] or (2) by accessing an unlocked workstation. Another vector which does not involve access to a restricted area, is to access to the internal network by hacking Wi-Fi access points. This can be accomplished using existing vulnerabilities such as 'Krack' [47] or the more recent 'Bleeding-Bit' vulnerabilities which have affected many hospitals [48].

Compromising the PACS. Once access to the PACS has been achieved, there are numerous ways an attacker can compromise a target asset. Aside from exploiting misconfigurations or default credentials, the attacker can exploit known software vulnerabilities. With regards to PACS servers, some already disclose private information/credentials which can be exploited remotely to create of admin accounts, and have hard-coded credentials.[8] A quick search on `exploit-db.com` reveals seven implemented exploits for PACS servers in 2018 alone. With regards to modality workstations, they too have been found to have significant vulnerabilities [49]. In 2018

[8] `CVE-2017-14008` and `CVE-2018-17906`

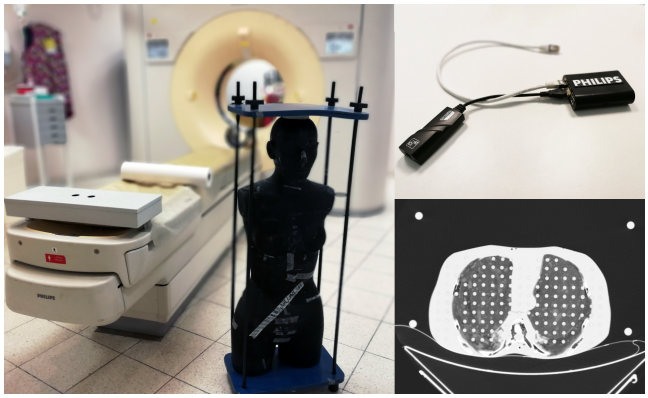

Figure 5: Left: The CT scanner and the medical dummy used to validate the attack. Top-right: the Pi-bridge used to intercept the scans. Bottom-right: one of the dummy's slices, sent by the CT scanner, and intercepted by the Pi-bridge.

the US Department of Homeland Security exposed 'low skill' vulnerabilities in Philips' Brilliance CT scanners [50]. For example, improper authentication, OS command injection, and hard-coded credentials.[9] Other recent vulnerabilities include hard-coded credentials.[10]

Given the state of health-care security, and that systems such as CT scanners are rarely given software updates [51], it is likely that these vulnerabilities and many more exist. Once the target asset in the PACS has been compromised, the attacker will be able to install the malware and manipulate the scans of target patients.

4.5 Attack Demonstration

To demonstrate how an attacker can access and manipulate CT scans, we performed a penetration test on a hospital's radiology department. The pen-test was performed with full permission of the participating hospital. To gain access to all CT scans, we performed a man-in-the-middle attack on the CT scanner using a Raspberry Pi 3B. The Pi was given a USB-to-Ethernet adapter, and was configured as a passive network bridge (without network identifiers). The Pi was also configured as a hidden Wi-Fi access point for backdoor access. We also printed a 3D logo of the CT scanner's manufacturer and glued it to the Pi to make it less conspicuous. The pen-test was performed as follows: First we waited at night until the cleaning staff opened the doors. Then we found the CT scanner's room and installed the Pi-bridge between the scanner's workstation and the PACs network (location #2 in Fig. 3). Finally, we hid the Pi-bridge under an access panel in the floor. The entire installation process took 30 seconds to complete. We were able to connect to the Pi wirelessly from the waiting room (approximately 20m away) to monitor the device's status, change the target identifier, etc.

At this point, an attacker could either intercept scans directly or perform lateral movement through the PACS to other subsystems and install the malware there. To verify that we could intercept and manipulate the scans, we scanned a medical dummy (Fig. 5). We found that the scan of the dummy was sent over the network twice: once in cleartext over TCP to an internal web viewing service, and again to the PACS storage server using TLSv1.2. However, to our surprise, the payload of the TLS transmission was also in cleartext. Moreover, within 10 minutes, we obtained the usernames and passwords of over 27 staff members and doctors due to multicasted Ethernet traffic containing `HTTP POST` messages sent in cleartext. A video of the pen-test can be found online.[11]

These vulnerabilities were disclosed to the hospital's IT staff and to their PACS software provider. Though inquiry, we found that it is not common practice for hospitals to encrypt their internal PACs traffic [52]. One reason is compatibility: hospitals often have old components (scanners, portals, databases, ...) which do not support encryption. Another reason is some PACS are not directly connected to the Internet, and thus is it erroneously thought that there is no need for encryption.

5 The CT-GAN Framework

In this section, we present the technique which an attacker can use to add/remove evidence in CT scans. First, we present the CT-GAN architecture and how to train it. Then, we will describe the entire tampering process and present some sample results. As a case study, we will focus on injecting/removing lung cancer.

It is important to note that there are many types of lung cancer. A common type of cancer forms a round mass of tissue called a solitary pulmonary nodule. Most nodules with a diameter less than 8mm are benign. However, nodules which are larger may indicate a malign cancerous growth. Moreover, if *numerous* nodules having a diameter $>$ 8mm are found, then the patient has an increased risk of primary cancer [53]. For this attack, we will focus on injecting and removing multiple solitary pulmonary nodules.

5.1 The Neural Architecture

A single slice in a CT scan has a resolution of *at least* 512x512 pixels. Each pixel in a slice measures the radiodensity at that location in Hounsfield units (HU). The CT scan of a human's lungs can have over 157 million voxels[12] (512x512x600). In order to train a GAN on an image of this size, we first locate a candidate location (voxel) and then cut out a small region around it (cuboid) for processing. The selected region is slightly larger than needed in order to provide the cGAN with context of the surrounding anatomy. This enables the cGAN to generate/remove lung cancers which connect to the body in a realistic manner.

To accurately capture the concepts of injection and removal, we use a framework consisting of two cGANs: one for injecting cancer (GAN_{inj}) and one for removing cancer (GAN_{rem}). Both GAN_{inj} and GAN_{rem} are deep 3D convolutional cGANs

[9] `CVE-2018-8853`, `CVE-2018-8857`, and `CVE-2018-8861`
[10] `CVE-2017-9656`
[11] `https://youtu.be/_mkRAArj-x0`
[12] A voxel is the three dimensional equivalent of a pixel.

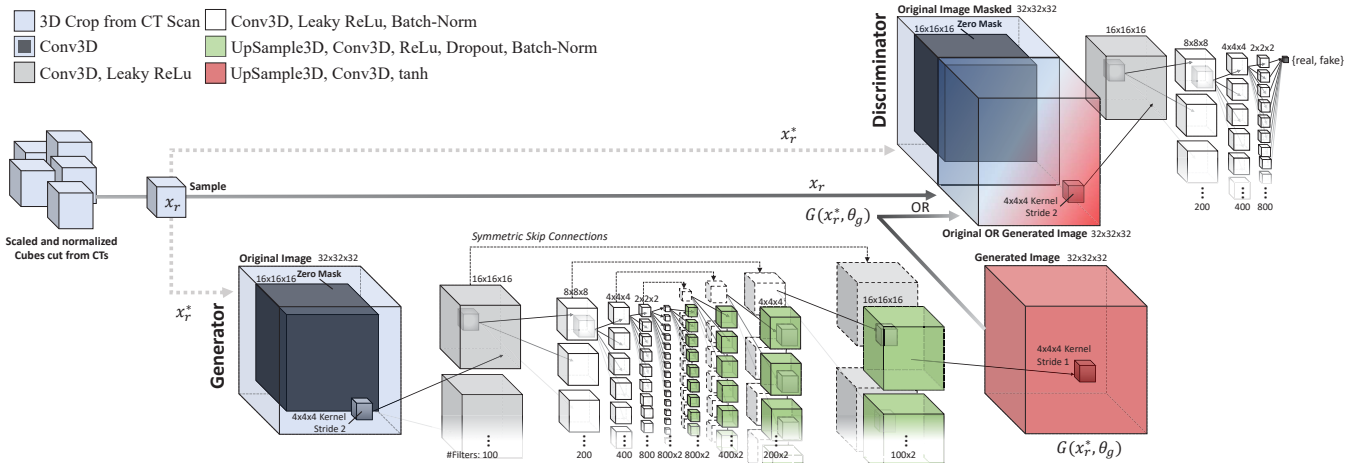

Figure 6: The network architecture, layers, and parameters used for both the injection (GAN_{inj}) and removal (GAN_{rem}) networks.

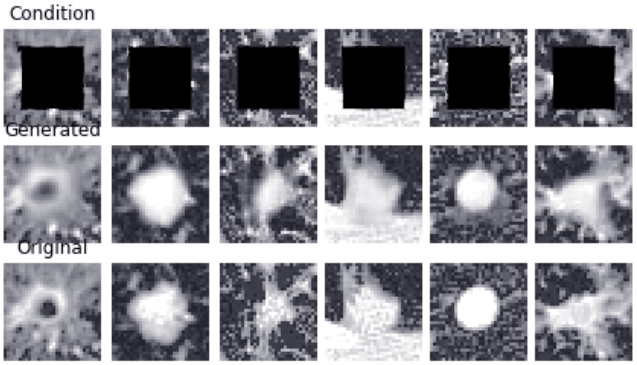

Figure 7: Training samples after 100 epochs showing the middle slice only. Top: the masked sample x_r^* given to both the *generator* G_{inj} and *discriminator* D_{inj}. Middle: The in-painted image x_g produced by the G_{inj}. Bottom: the ground-truth x_r. Note, D_{inj} sees either (x_r^*, x_r) or (x_r^*, x_g).

trained to perform in-painting on samples which are 32^3 voxels in dimension. For the completion mask, we zero-out a 16^3 cube in the center of the input sample. To inject a large pulmonary nodule into a CT scan, we train GAN_{inj} on cancer samples which have a diameter of least 10mm. As a result, the trained *generator* completes sample cuboids with similar sized nodules. To remove cancer, GAN_{rem} is trained using the same architecture, but with samples containing benign lung nodules only (having a diameter <3mm).

The model architecture (layers and configurations) used for both GAN_{inj} and GAN_{rem} is illustrated in Fig. 6. Overall, θ_g and θ_d had 162.6 million and 26.9 million trainable parameters respectively (189.5 million in total).

We note that follow up CT scans are usually ordered when a large nodule is found. This is because nodule growth is a strong indicator of cancer [53]. We found that an attacker is able to simulate this growth by conditioning each cancerous training sample on the nodule's diameter. However, the objective of this paper is to show how GANS can add/remove evidence realistically. Therefore, for the sake of simplicity, we have omitted this 'feature' from the above model.

5.2 Training CT-GAN

To train the GANs, we used a free dataset of 888 CT scans collected in the LIDC-IDRI lung cancer screening trial [54]. The dataset came with annotations from radiologists: the locations and diameters of pulmonary nodules having diameters greater than 3mm. In total there were 1186 nodules listed in the annotations.

To create the training set for GAN_{inj}, we extracted from the CT scans all nodules with a diameter between 10mm and 16mm (169 in total). To increase the number of training samples, we performed data augmentation: For each of the 169 cuboid samples, we (1) flipped the cuboid on the x, y, and xy planes, (2) shifted the cuboid by 4 pixels in each direction on the xy plane, and (3) rotated the cuboid 360 degrees at 6 degree intervals. This produced an additional 66 instances for each sample. The final training set had 11,323 training samples.

To create the training set for GAN_{rem}, we first selected clean CT scans which had no nodules detected by the radiologists. On these scans, we used the nodule detection algorithm from [55] (also provided in the dataset's annotations) to find benign micro nodules. Of the detected micro nodules, we selected 867 nodules at random and performed the same data augmentation as above. The final training set had 58,089 samples.

Prior to training the GANs, all of the samples were preprocessed with scaling, equalization, and normalization (described in the next section in detail). Both of the GANs were trained on their respective datasets for 200 epochs with a batch size of 50 samples. Each GAN took 26 hours to complete its training on an NVIDIA GeForce GTX TITAN X using all of the GPU's memory. Fig. 7 shows how well GAN_{inj} was able to in-paint cancer patterns after 150 epochs.

5.3 Execution: The Tampering Process

In order to inject/remove lung cancer, pre/post-processing steps are required. The following describes the entire

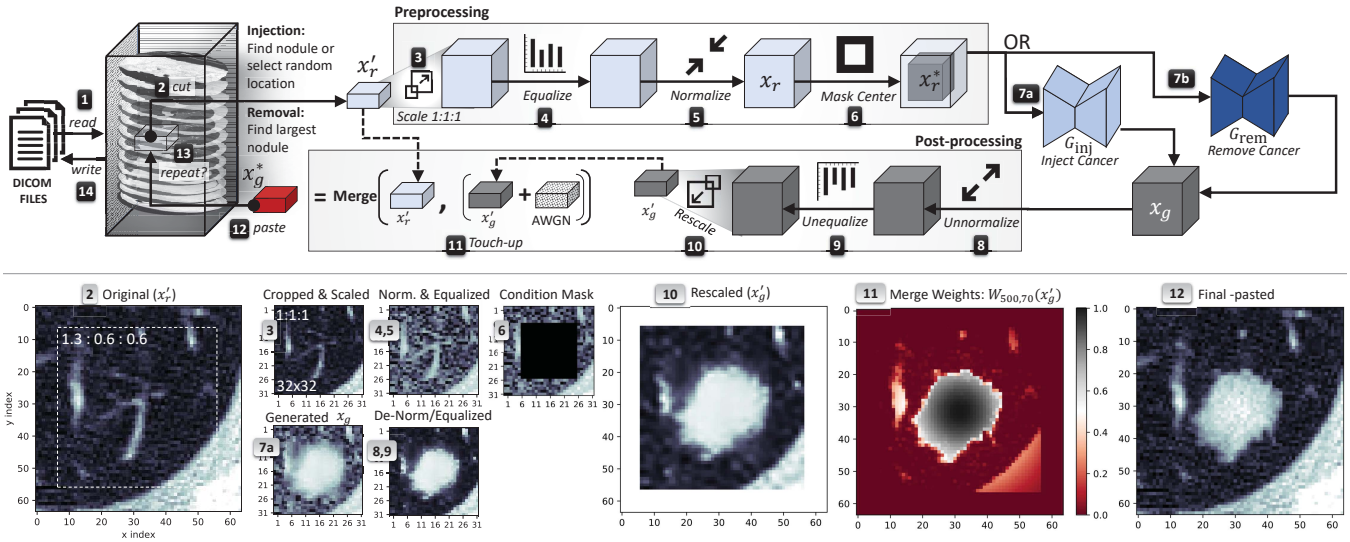

Figure 8: Top: the complete cancer injection/removal process. Bottom: sample images from the *injection* process. The grey numbers indicate from which step the image was taken. The sample 2D images are the middle slice of the respective 3D cuboid.

injection/removal process as illustrated in Fig. 8:

1. **Capture Data.** The CT scan is captured (as data-at-rest or data-in-motion) in either raw or DICOM format using one of the attack vectors from section 4.

2. **Localize & Cut.** A candidate location is selected where cancer will be injected/removed, and then the cuboid x'_r is cut out around it.

   Injection: An injection location can be selected in one of two ways. The fastest way is to take one of the middle slices of the CT scan and select a random location near the middle of the left or right half (see Fig. 16 in the appendix). With 888 CT scans, this strategy gave us a 99.1% successes rate. A more precise way is to execute an existing nodule detection algorithm to find a random micro nodule. To improve speed, the algorithm can be given only a few slices and implemented with early stopping. In our evaluation, we used the algorithm in [55], though many other options are available.

   Removal: A removal location can be found by selecting the largest nodule with [55], or by using a pre-trained deep learning cancer detection model.[13]

3. **Scale.** x'_r is scaled to the original 1:1:1 ratio using 3D spline interpolation.[14] The ratio information is available in the DICOM meta data with the tags (`0x0028`,`0x0030`) and (`0x0018`,`0x0050`). Scaling is necessary because each scan is stored with a different aspect ratio, and a GAN needs consistent units to produce accurate results. To minimize the computations, the cuboid cut in step 2 is cut with the exact dimensions so that the result of the rescaling process produces a 32^3 cube.

4-5. **Equalize & Normalize.** Histogram equalization is applied to the cube to increase contrast. This is a critical step since it enables the GAN to learn subtle features in the anatomy Normalization is then applied using the formula $X_n = 2\frac{X-\min(X)}{\max(X)-\min(X)} - 1$. This normalization ensures that all values fall on the range of $[-1,1]$ which helps the GAN learn the features better. The output of this process is the cube x_r.

6. **Mask.** In the center of x_r, a 16^3 cube is masked with zeros to form x^*_r. The masked area will be in-painted (completed) by the *generator*, and the unmasked area is the context).

7. **Inject/Remove.** x^*_r is passed through the chosen *discriminator* (G_{inj} or G_{rem}) creating a new sample (x_g) with new 3D generated content.

8-10. **Reverse Preprocessing.** x_g is unnormalized, unequalized, and then rescaled with spline interpolation back to its original proportions, forming x'_g.

11. **Touch-up.** The result of the interpolation usually blurs the imagery. In order to hide this artifact from the radiologists, we added Gaussian noise to the sample: we set $\mu = 0$ and σ to the sampled standard deviation of x'_r. To get a clean sample of the noise, we only measured voxels with values less than -600 HU. Moreover, to copy the relevant content into the scan, we merged the original cuboid (x'_r) with the generated one (x'_g) using a sigmoid weighted average.

    Let W be the weight function defined as

$$W_{\alpha,\beta}(x) = \frac{1}{1+e^{-(x+\alpha)/\beta}} * G(x) \qquad (1)$$

    where parameter α is the HU threshold between wanted and unwanted tissue densities, and parameter β controls the smoothness of the cut edges. The function $G(x)$ returns a 0-1 normalized Gaussian kernel with the dimensions of

[13] Pre-trained models are available here: https://concepttoclinic.drivendata.org/algorithms

[14] In Python: `scipy.ndimage.interpolation.zoom`

x. $G(x)$ is used to decay the contribution of each voxel the further it is the cuboid's center.

With W, we define the merging function as

$$merge_{\alpha,\beta}(x,y) = W_{\alpha,\beta}(x) * x + \left(1 - W_{\alpha,\beta}(x)\right) * y \quad (2)$$

where x is source (x'_g) and y is the destination (x'_r). We found that setting $\alpha = 500$ and $\beta = 70$ worked best. By applying these touch-ups, the final cuboid x^*_g is produced.

12. **Paste.** The cuboid x^*_g is pasted back into the CT scan at the selected location. See Fig. 16 in the appendix for one slice of a complete scan.

13. **Repeat.** If the attacker is removing cancer, then return to step 2 until no more nodules with a diameter $>$ 3mm are found. If the attacker is injecting cancer, then (optionally) return to step 2 until four injections have been performed. The reason for this is because the risk of a patient being diagnosed with cancer is statistically greater in the presence of exactly four solitary pulmonary nodules having a diameter $>$ 8mm [53].

14. **Return Data.** The scan is converted back into the original format (e.g. DICOM) and returned back to the source.

The quality of the injection/removal process can be viewed in figures 9 and 10. Fig. 9 presents a variety of examples before and after tampering, and Fig. 10 provides a 3D visualization of a cancer being injected and removed. More visual samples can be found in the appendix (figures 19 and 20).

We note that although some steps involve image touch-ups, the entire process is automatic (unlike Photoshop) and thus can be deployed in an autonomous malware or inside a viewing application (real-time tampering). We note that the same neural architecture and tampering process works on other modalities and medical conditions. For example, Fig. 18 in the appendix shows CT-GAN successfully injecting brain tumors into MRI head scans.

6 Evaluation

In this section we present our evaluation on how well the CT-GAN attack can fool expert radiologists and state-of-the-art AI.

6.1 Experiment Setup

To evaluate the attack, we recruited three radiologists (denoted **R1**, **R2**, and **R3**) with 2, 5, and 7 years of experience respectively. We also used a trained lung cancer screening model (denoted **AI**), the same deep learning model which won the 2017 Kaggle Data Science Bowl (a $1 million competition for diagnosing lung cancer).[15]

The experiment was performed in two trials: blind and open. In the blind trial, the radiologists were asked to diagnose 80 *complete* CT scans of lungs, but they were not told the purpose of the experiment or that some of the scans were manipulated.

[15] Source code and model available here: https://github.com/lfz/DSB2017

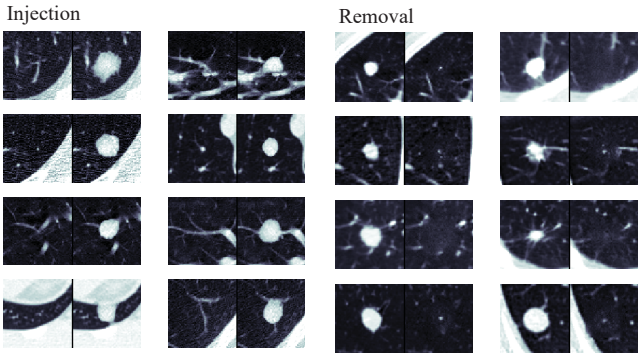

Figure 9: Sample injections (left) and removals (right). For each image, the left side is before tampering and the right side is after. Note that only the middle 2D slice is shown.

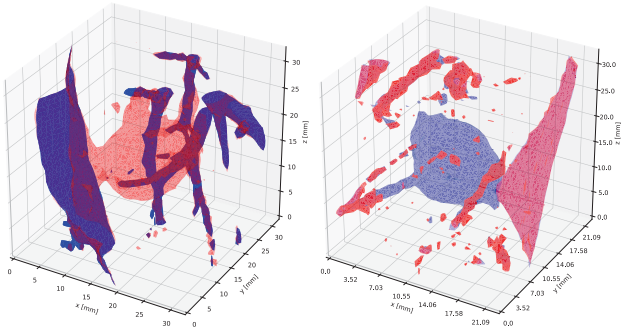

Figure 10: A 3D sample of injection (left) and removal (right) before (blue) and after (red) tampering with the CT scan.

Table 2: Summary of the scans and the relevant notations

	# Scans				# Slices	# Nodules		
	TB	TM	FB	FM		TM	FB	FM
Blind Trial	10	10	30	30	17,941	23	61	34
Open Trial	5	5	5	5	5,296	12	11	7

True-Benign (TB): *an original instance with no cancer*
True-Malign (TM): *an original instance with cancer*
False-Benign (FB): *a tampered instance with cancer removed*
False-Malign (FM): *a tampered instance with cancer injected*

In the open trial, the radiologists were told about the attack, and were asked to identify fake, real, and removed nodules in 20 CT scans. In addition, the radiologists were asked to rate the confidence of their decisions. After each trial, we gave the radiologists a questionnaire to assess how susceptible they were to the attacks. In all cases, the radiologists were asked to only detect and diagnose pulmonary nodules which have a diameter greater than 3mm.

The CT scans were taken from the LIDC-IDRI dataset [54]. The set of CT scans used in each trial and the notations used in this section are available in Table 2.

False benign (FB) and true malign (TM) scans truthfully contained at least one nodule with a diameter between 10mm and 16mm. FB scans were made by removing every nodule in the scan. FM scans were made by randomly injecting 1-4

Table 3: Cancer Detection Performance - Blind Trial

Evaluated by...		*Detector*																													
		Radiologist #1 (R1)								**Radiologist #2 (R2)**								**Radiologist #3 (R3)**								**Artificial Intelligence (AI)**							
		FP	TP	FN	TN	FPR	TPR	FNR	TNR	FP	TP	FN	TN	FPR	TPR	FNR	TNR	FP	TP	FN	TN	FPR	TPR	FNR	TNR	FP	TP	FN	TN	FPR	TPR	FNR	TNR
Instance	FB	0	2	59	0	-	0.03	0.97	-	0	3	58	0	-	0.05	0.95	-	0	4	57	0	-	0.07	0.93	-	0	4	57	0	-	0.07	0.93	-
	FM	32	0	0	2	0.94	-	-	0.06	33	0	0	1	0.97	-	-	0.03	33	0	0	1	0.97	-	-	0.03	34	0	0	0	1.00	-	-	0.00
	TB	0	0	0	15	0.00	-	-	1.00	0	0	0	15	0.00	-	-	1.00	0	0	0	15	0.00	-	-	1.00	0	0	0	15	0.00	-	-	1.00
	TM	0	21	2	0	-	0.91	0.09	-	0	19	4	0	-	0.83	0.17	-	0	20	3	0	-	0.87	0.13	-	0	21	2	0	-	0.91	0.09	-
Patient	FB	0	2	28	0	-	0.07	0.93	-	0	3	27	0	-	0.10	0.90	-	0	4	26	0	-	0.13	0.87	-	0	4	26	0	-	0.13	0.87	-
	FM	29	0	0	1	0.97	-	-	0.03	30	0	0	0	1.00	-	-	0.00	30	0	0	0	1.00	-	-	0.00	30	0	0	0	1.00	-	-	0.00
	TB	0	0	0	10	0.00	-	-	1.00	0	0	0	10	0.00	-	-	1.00	0	0	0	10	0.00	-	-	1.00	0	0	0	10	0.00	-	-	1.00
	TM	0	9	1	0	-	0.90	0.10	-	0	10	0	0	-	1.00	0.0	-	0	10	0	0	-	1.00	0.00	-	0	10	0	0	-	1.00	0.00	-

Table 4: Attack Detection Confusion Matrix - Open Trial
Evalauted by Instance

Seemingly Benign	**R1**		**R2**		**R3**	
	FB	TB	FB	TB	FB	TB
Cancer Removed: FB	0	11	2	7	0	10
No Cancer: TB	1	6	0	6	4	6

Seemingly Malign	**R1**		**R2**		**R3**	
	FM	TM	FM	TM	FM	TM
Fake Cancer: FM	0	3	5	1	3	4
Real Cancer: TM	2	6	3	6	2	4

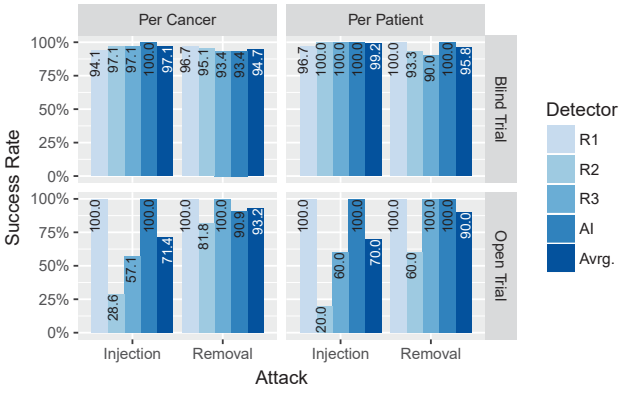

Figure 11: Attack success rates - Both Trials.

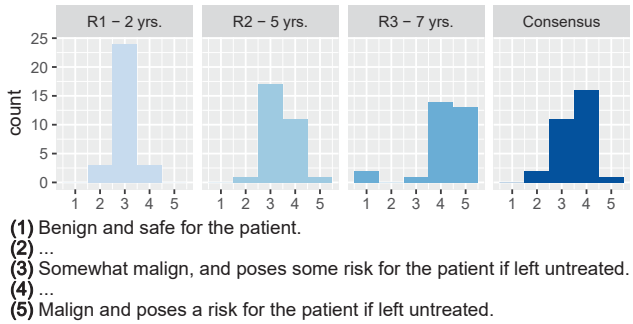

Figure 12: Malignancy of injected cancers (FM) - Blind Trial.

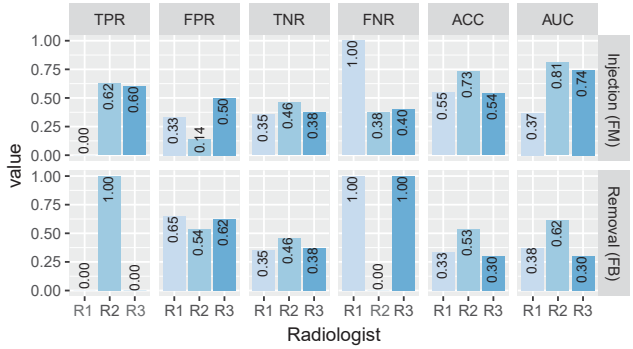

Figure 13: Attack detection performance - Open Trial.

nodules into a benign scan, where the injected nodules had a diameter of 14.4mm on average. In total, there were 100 CT scans analyzed by each of the radiologists, and the radiologists spent approximately 10 minutes analyzing each of these scans.

We note that the use of three radiologists is common practice in medical research (e.g., [56]). Moreover, we found that radiologists (and AI) significantly agreed with each other's diagnosis per patient and per nodule. We verified this agreement by computing Fliess' kappa [57] (a statistical interrater reliability measure) which produced an excellent kappa of 0.84 (p-value < 0.0001). Therefore, adding more radiologists will likely not affect the results.

6.2 Results: Blind Trial

In Table 3 we present the cancer detection performance of the radiologists and AI. The table lists the number of false-positives (FP - *detected a non-existent cancer*), true-positives (TP - *detected a real cancer*), false-negatives (FN - *missed a real cancer*), and their respective rates. The TCIA annotations (nodule locations) were used as our ground truth for measuring the performance on FB and TM scans. We evaluated these metrics per instance of cancer, and per patient as a whole. All four detectors performed well on the baseline (TB and TM) having an average TPR of 0.975 and a TNR of 1.0 in diagnosing the patients, meaning that we can rely on their diagnosis.

The top of Fig. 11 summarizes the attack success rates for the blind trial. In general, the attack had an average success rate of 99.2% for cancer injection and 95.8% for cancer removal. The AI was fooled completely which is an important aspect since some radiologists use AI tools to support their analysis (e.g. the Philips IntelliSite Pathology Solution). The radiologists were fooled less so, primarily due to human error (e.g., missing a nodule). When asked, none of the radiologists reported anything abnormal with the scans with the exception

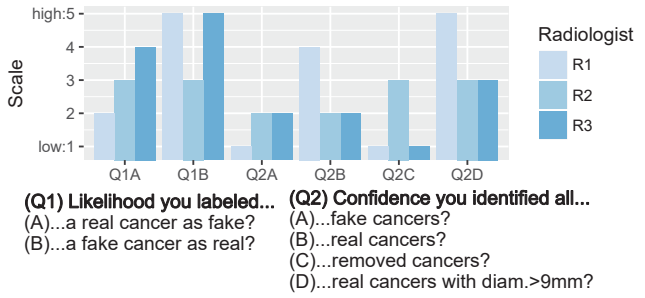

Figure 14: Confidence in detecting attacks - Open Trial.

of R2 who noted some noise in the area of one removal (FB). This may be attributed to "inattentional blindness," where one may miss an obvious event (artifacts) while engaged in a different task (searching for large nodules). In [58], the authors showed that this phenomenon also affects radiologists.

With regards to the injected cancers (FM), the consensus among the radiologists was that one-third of the injections require an immediate surgery/biopsy, and that all of the injections require follow-up treatments/referrals. When asked to rate the overall malignancy of the FM patients, the radiologists said that nearly all cases were significantly malign and pose a risk to the patient if left untreated. Fig. 12 summarizes radiologists' ratings of the FM patients. One interesting observation is that the malignancy rating increased with the experience of the radiologist. Finally, we note that an attacker could increase the overall malignancy of the injections if CT-GAN were trained only on samples with high malignancy and/or a larger diameter.

6.3 Results: Open Trial

In Table 4 we present the radiologists' attack detection performance with knowledge of the attack. Fig. 13 summarizes these results and provides the radiologists' accuracy (ACC) and area under the curve (AUC). An AUC of 1.0 indicates a perfect binary classifier, whereas an AUC of 0.5 indicates random guessing. The results show that the radiologists could not consistently tell the difference between real and fake cancers or identify the locations of removed cancers.

With regards to the attack success rates (bottom of Fig. 11), knowledge of the attack did not significantly affect cancer removal (90% from 95.8%). However, the success of the cancer injection was affected (70% from 99.2%). Moreover, R2 also picked up on a particular pattern which gave away several instances. This is a promising result, since it indicates that a portion of CT-GAN's attacks can be mitigated by educating radiologists. However, aside from low accuracy (61% for detecting an injection and 39% for detecting a removal), there was a significant number of false positives. With a high likelihood of making a mistake, a radiologist may choose not to report abnormalities. This is also apparent from the low confidence scores which the radiologists gave decisions (see Fig. 14).

In summary, both radiologists and AI are highly susceptible to CT-GAN's image tampering attacks. Although knowledge of the attack can help mitigate some cases of cancer injection, the error rates and confidence scores suggest that the attack may not be reported in most cases.

6.4 Baseline Analysis

As mentioned earlier, the attacker cannot use Photoshop on the images. This may be due to a time restriction: the radiologist may perform the diagnosis immediately after the scan has been taken, or the malware may be in the viewing application and operating in real-time. This may also be due to a connectivity restriction: many PACS do not have a direct connection to the Internet. Therefore, the attacker would need to infect the PACS with a malware that can perform the tampering autonomously.

However, for cancer injection, one may argue that there are simpler *automated* methods than using a GAN. For example, the malware could simply paste a pre-saved cropped image of cancer into a scan at a candidate location (found using steps 1 and 2 in Fig. 8). Therefore, to validate the use of a GAN, we evaluated this splice attack as a baseline comparison.

The experiment was setup as follows. To create our set of 'pre-saved cancers' we carefully cut out ten 32x32x32 cubes of real cancerous nodules from [54]. We then removed the irrelevant background tissue by producing transparency (alpha) channels using equation (1).

The evaluation dataset contained 20 scans in total: 10 FM and 10 TM. To create a FM scan, we took a random healthy scan and used steps 1 and 2 of Fig. 8 to select a candidate injection location. We then pasted one of the pre-saved cancers into the scan at that location using the transparency channel. To create the FM scans, we randomly selected 10 sick patients with exactly one nodule each.

To evaluate the baseline attack, we asked radiologist **R2** to analyze the 20 scans and determine which of the scans were authentic (TM) and which were tampered (FM) –similar to the open trial. This task took the radiologist approximately two hours to perform.

In summary, **R2** correctly labeled all of the tampered scans as fake. The only error was that **R2** mislabeled three of the authentic scans as fake. These results indicate that the baseline cancer injection attack cannot trick a radiologist, in comparison to CT-GAN which succeeds nearly every time.

The reason the baseline attack failed to trick **R2** is because the process of automatically pasting cancer creates obvious artifacts. This is because the pasted samples ignore the surrounding anatomy and may contain inconsistent noise patterns (textures). Fig. 15 illustrates some examples of these abnormalities such as cut bronchi, inconsistent noise patterns, and unaligned borders. CT-GAN does not produce these artifacts because it uses in-painting which considers the original content and surrounding anatomy.

7 Countermeasures

The tampering of DICOM medical files is a well-known concern. In the section we provide a brief overview of solutions for preventing and detecting this attack.

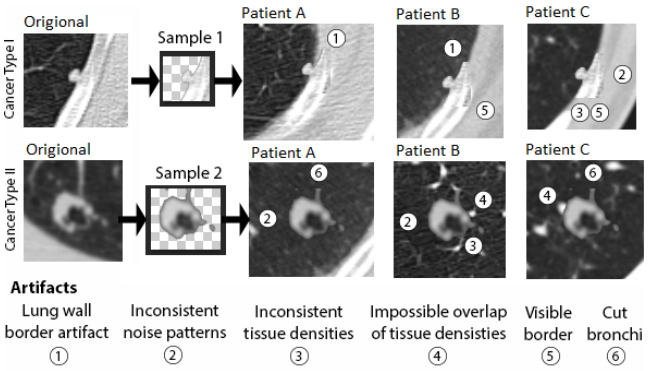

Figure 15: An illustration showing artifacts which can occur when using an *unsupervised* splice attack instead of CT-GAN. Only the middle slice is shown.

7.1 Prevention

To mitigate this threat, administrators should secure both the data-in-motion (DiM) and the data-at-rest (DaR). To secure data-in-motion, admins should enable encryption between the hosts in their PACS network using proper SSL certificates. This may seem trivial, but after discovering this flaw in the hospital we pen-tested, we turned to the PACS software provider for comment. The company, with over 2000 installations worldwide, confirmed to us that their hospitals do not enable encryption in their PACS because "it is not common practice". We were also told that some of the PACS don't support encryption at all.[16] To secure the DaR, servers and anti-virus software on modality and radiologist workstations should be kept up to date, and admins should also limit the exposure which their PACS server has to the Internet.

7.2 Detection

The best way to detect this attack is to have the scanner sign each scan with a digital signature. The DICOM image file standard already allows users to store signatures within the file's data structure [59, 60]. However, although some PACS software providers offer this feature, we have not seen it in use within a PACS. If enabled, admins should check that valid certificates are being used and that the radiologists' viewing applications are indeed verifying the signatures.

Another method for detecting this attack is digital watermarking (DW). A DW is a hidden signal embedded into an image such that tampering corrupts the signal and thus indicates a loss of integrity. For medical images, this subject has been researched in depth [20] and can provide a means for localizing changes in a tampered image. However, we did not find any medical devices or products which implement DW techniques. This may be due to the fact that they add noise to images which may harm the medical analysis.

Tampered images can also be detected with machine learning. In the supervised setting (where models are trained on examples of tampered images) the authors in [61] propose detection by (1) extracting a scan's noise pattern using a Wiener filter, then (2) applying a multi-resolution regression filter on the noise, and then (3) executing an SVM and ELM together via a Bayesian Sum Rule model. Many domain specific methods exist for detecting images tampered by GANs (e.g., images/videos of faces [62–64]). However, the supervised approach in [65] is more suitable for detecting our attack since it is domain generic.

Several approaches have been proposed for unsupervised setting as well. These approaches attempt to detect anomalies (inconsistencies) within the tampered images. To detect these inconsistencies, researchers have considered JPEG blocks, signal processing, and compression/resampling artifacts [66]. For example, in a recent work the authors trained a Siamese network to predict the probability that a pair of patches from two images have the same EXIF metadata (e.g., focal length and shutter speed) [67]. In [67], the model is trained using a dataset of real images only. In [68], the authors proposed 'noiseprint' which uses a Siamese network to extract the camera's unique noise pattern from an image (PRNU) to find inconsistent areas. In their evaluation, the authors show that they can detect GAN-based inpainting. In [69], the authors proposed three strategies for using PRNU-based tampering localization techniques with multi-scale analysis. Using this method, the authors were able to detect forgeries of all shapes and sizes.

While these countermeasures may apply to CT-GAN in some cases, they do admit some caveats; namely, that (1) medical scans are usually not compressed so compression methods are irrelevant, (2) these methods were tested on 2D images and not 3D volumetric imagery, and (3) CT/MR imaging systems produce very different noise patterns than standard cameras. For example, we found that the PRNU method in [69] does not work out-of-the-box on our tampered CT scans. This is because the noise patterns of CT images are altered by a radon transform used to construct the image. As future work, we plan to research how these techniques can be applied to detecting attacks such as CT-GAN.

8 Conclusion

In this paper we introduced the possibility of an attacker modifying 3D medical imagery using deep learning. We explained the motivations for this attack, discussed the attack vectors (demonstrating one of them), and presented a manipulation framework (CT-GAN) which can be executed by a malware autonomously. As a case study, we demonstrated how an attacker can use this approach to inject or remove lung cancer from full resolution 3D CT scans using free medical imagery from the Internet. We also evaluated the attack and found that CT-GAN can fool both humans and machines: radiologists and state-of-the-art AI. This paper also demonstrates how we should be wary of closed world assumptions: both human experts and advanced AI can be fooled if they fully trust their observations.

[16] See [52] for further comments.

References

[1] P. I, W. LR, et al. Health care spending in the united states and other high-income countries. *JAMA*, 319(10):1024–1039, 2018.

[2] J. R. Haaga. *CT and MRI of the Whole Body*. No. v. 1 in CT and MRI of the Whole Body. Mosby/Elsevier, 2008. ISBN 9780323053754.

[3] H. I. News. The biggest healthcare data breaches of 2018 (so far). https://www.healthcareitnews.com/projects/biggest-healthcare-data-breaches-2018-so-far, 2019.

[4] T. George. Feeling the pulse of cyber security in healthcare, securityweek.com. https://www.securityweek.com/feeling-pulse-cyber-security-healthcare, 2018.

[5] I. Institute. Cybersecurity in the healthcare industry. https://resources.infosecinstitute.com/cybersecurity-in-the-healthcare-industry, 2016.

[6] L. Coventry and D. Branley. Cybersecurity in healthcare: A narrative review of trends, threats and ways forward. *Maturitas*, 113:48 – 52, 2018. ISSN 0378-5122.

[7] M. S. Jalali and J. P. Kaiser. Cybersecurity in hospitals: A systematic, organizational perspective. *Journal of medical Internet research*, 20(5), 2018.

[8] C. Beek. Mcafee researchers find poor security exposes medical data to cybercriminals, mcafee blogs. https://securingtomorrow.mcafee.com/other-blogs/mcafee-labs/mcafee-researchers-find-poor-security-exposes-medical-data-to-cybercriminals/, 2018.

[9] H. Huang. *PACS-Based Multimedia Imaging Informatics: Basic Principles and Applications*. Wiley, 2019. ISBN 9781118795736.

[10] Verizon. Protected health information data breach report. *white paper*, 2018.

[11] F. Bray, J. Ferlay, et al. Global cancer statistics 2018: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: a cancer journal for clinicians*, 68(6):394–424, 2018.

[12] X. Wu, K. Xu, et al. A survey of image synthesis and editing with generative adversarial networks. *Tsinghua Science and Technology*, 22(6):660–674, 2017.

[13] I. Goodfellow, J. Pouget-Abadie, et al. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680. 2014.

[14] W. Hu and Y. Tan. Generating adversarial malware examples for black-box attacks based on gan. *arXiv preprint arXiv:1702.05983*, 2017.

[15] M. Rigaki and S. Garcia. Bringing a gan to a knife-fight: Adapting malware communication to avoid detection. In *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 70–75. IEEE, 2018.

[16] R. Chesney and D. K. Citron. Deep fakes: A looming challenge for privacy, democracy, and national security. *U of Texas Law, Public Law Research Paper No. 692; U of Maryland Legal Studies Research Paper No. 2018-21*, 2018.

[17] P. Isola, J.-Y. Zhu, et al. Image-to-image translation with conditional adversarial networks. *arXiv preprint*, 2017.

[18] T. Seals. Rsa conference 2019: Ultrasound hacked in two clicks, threatpost. https://threatpost.com/ultrasound-hacked/142601/, 2019.

[19] J.-Y. Zhu, T. Park, et al. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint*, 2017.

[20] A. K. Singh, B. Kumar, et al. *Medical Image Watermarking Techniques: A Technical Survey and Potential Challenges*, pp. 13–41. Springer International Publishing, Cham, 2017. ISBN 978-3-319-57699-2.

[21] S. Sadeghi, S. Dadkhah, et al. State of the art in passive digital image forgery detection: copy-move image forgery. *Pattern Analysis and Applications*, 21(2):291–306, May 2018. ISSN 1433-755X.

[22] A. Kharboutly, W. Puech, et al. Ct-scanner identification based on sensor noise analysis. In *2014 5th European Workshop on Visual Information Processing (EUVIP)*, pp. 1–5. Dec 2014.

[23] Y. Duan, D. Bouslimi, et al. Computed tomography image origin identification based on original sensor pattern noise and 3d image reconstruction algorithm footprints. *IEEE journal of biomedical and health informatics*, 21(4):1039–1048, 2017.

[24] X. Yi, E. Walia, et al. Generative adversarial network in medical imaging: A review. *arXiv preprint arXiv:1809.07294*, 2018.

[25] L. Bi, J. Kim, et al. Synthesis of Positron Emission Tomography (PET) Images via Multi-channel Generative Adversarial Networks (GANs). pp. 43–51. Springer, Cham, 2017.

[26] A. Ben-Cohen, E. Klang, et al. Virtual PET Images from CT Data Using Deep Convolutional Networks: Initial Results. pp. 49–57. Springer, Cham, 2017.

[27] —. Cross-Modality Synthesis from CT to PET using FCN and GAN Networks for Improved Automated Lesion Detection. 2 2018.

[28] Q. Dou, C. Ouyang, et al. Unsupervised Cross-Modality Domain Adaptation of ConvNets for Biomedical Image Segmentations with Adversarial Loss. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, pp. 691–697. International Joint Conferences on Artificial Intelligence Organization, California, 7 2018. ISBN 9780999241127.

[29] C.-B. Jin, H. Kim, et al. Deep CT to MR Synthesis using Paired and Unpaired Data. 5 2018.

[30] C. Bermudez, A. J. Plassard, et al. Learning implicit brain mri manifolds with deep learning. In *Medical Imaging 2018: Image Processing*, vol. 10574, p. 105741L. International Society for Optics and Photonics, 2018.

[31] M. Frid-Adar, I. Diamant, et al. GAN-based Synthetic Medical Image Augmentation for increased CNN Performance in Liver Lesion Classification. 3 2018.

[32] J. M. Wolterink, T. Leiner, et al. Blood Vessel Geometry Synthesis using Generative Adversarial Networks. In *1st Conference on Medical Imaging with Deep Learning (MIDL 2018)*. Amsterdam, The Netherlands, The Netherlands, 2018.

[33] C. Baur, S. Albarqouni, et al. Melanogans: High resolution skin lesion synthesis with gans. *arXiv preprint arXiv:1804.04338*, 2018.

[34] A. Madani, M. Moradi, et al. Chest x-ray generation and data augmentation for cardiovascular abnormality classification. In *Medical Imaging 2018: Image Processing*, vol. 10574, p. 105741M. International Society for Optics and Photonics, 2018.

[35] M. J. Chuquicusma, S. Hussein, et al. How to fool radiologists with generative adversarial networks? a visual turing test for lung cancer diagnosis. In *Biomedical Imaging (ISBI 2018), 2018 IEEE 15th International Symposium on*, pp. 240–244. IEEE, IEEE, 4 2018. ISBN 978-1-5386-3636-7.

[36] W. Hruby. *Digital (R)Evolution in Radiology*. Springer Vienna, 2013. ISBN 9783709137079.

[37] A. Peck. *Clark's Essential PACS, RIS and Imaging Informatics*. Clark's Companion Essential Guides. CRC Press, 2017. ISBN 9781498763462.

[38] C. Carter and B. Veale. *Digital Radiography and PACS*. Elsevier Health Sciences, 2018. ISBN 9780323547598.

[39] B. Siwicki. Cloud-based pacs system cuts imaging costs by half for rural hospital | healthcare it news. https://www.healthcareitnews.com/news/cloud-based-pacs-system-cuts-imaging-costs-half-rural-hospital.

[40] J. Bresnick. Picture archive communication system use widespread in hospitals. https://healthitanalytics.com/news/picture-archive-communication-system-use-widespread-in-hospitals, 2016.

[41] S. Jodogne, C. Bernard, et al. Orthanc-a lightweight, restful dicom server for healthcare and medical research. In *Biomedical Imaging (ISBI), 2013 IEEE 10th International Symposium on*, pp. 190–193. IEEE, 2013.

[42] C. Costa, C. Ferreira, et al. Dicoogle-an open source peer-to-peer pacs. *Journal of digital imaging*, 24(5):848–856, 2011.

[43] L. Adefala. Healthcare experiences twice the number of cyber attacks as other industries. https://www.fortinet.com/blog/business-and-technology/healthcare-experiences-twice-the-number-of-cyber-attacks-as-othe.html, 2018.

[44] J. B. Rebecca Weintraub. 11 things the health care sector must do to improve cybersecurity. https://hbr.org/2017/06/11-things-the-health-care-sector-must-do-to-improve-cybersecurity, 2017.

[45] C. Osborne. Us hospital pays $55,000 to hackers after ransomware attack | zdnet. https://www.zdnet.com/article/us-hospital-pays-55000-to-ransomware-operators/, 2018.

[46] J. Muniz and A. Lakhani. *Penetration testing with raspberry pi*. Packt Publishing Ltd, 2015.

[47] M. Vanhoef and F. Piessens. Key reinstallation attacks: Forcing nonce reuse in wpa2. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1313–1328. ACM, 2017.

[48] A. NG. Security researchers find flaws in chips used in hospitals, factories and stores - cnet. https://www.cnet.com/news/security-researchers-find-flaws-in-chips-used-in-hospitals-factories-and-stores/, 2018.

[49] R. M. Robin Henry and J. Corke. Hospitals to struggle for days | news | the sunday times. https://www.thetimes.co.uk/article/nhs-cyberattack-bitcoin-wannacry-hospitals-to-struggle-for-days-k0nhk7p2b, 2017.

[50] DHS. Philips isite/intellispace pacs vulnerabilities (update a), ics-cert. https://ics-cert.us-cert.gov/advisories/ICSMA-18-088-01, 2018.

[51] J. E. Dunn. Imagine you're having a ct scan and malware alters the radiation levels – it's doable • the register. https://www.theregister.co.uk/2018/04/11/hacking_medical_devices/, 2018.

[52] K. Zetter. Hospital viruses: Fake cancerous nodes in ct scans, created by malware, trick radiologists. https://www.washingtonpost.com/technology/2019/04/03/hospital-viruses-fake-cancerous-nodes-ct-scans-created-by-malware-trick-radiologists/, April 2019.

[53] H. MacMahon, D. P. Naidich, et al. Guidelines for management of incidental pulmonary nodules detected on ct images: from the fleischner society 2017. *Radiology*, 284(1):228–243, 2017.

[54] S. G. Armato III, G. McLennan, et al. The lung image database consortium (lidc) and image database resource initiative (idri): a completed reference database of lung nodules on ct scans. *Medical physics*, 38(2):915–931, 2011.

[55] K. Murphy, B. van Ginneken, et al. A large-scale evalua-

tion of automatic pulmonary nodule detection in chest ct using local image features and k-nearest-neighbour classification. *Medical image analysis*, 13(5):757–770, 2009.

[56] A. Esteva, B. Kuprel, et al. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115, 2017.

[57] A. J. Conger. Integration and generalization of kappas for multiple raters. *Psychological Bulletin*, 88(2):322, 1980.

[58] T. Drew, M. L.-H. Võ, et al. The invisible gorilla strikes again: Sustained inattentional blindness in expert observers. *Psychological science*, 24(9):1848–1853, 2013.

[59] F. Cao, H. Huang, et al. Medical image security in a hipaa mandated pacs environment. *Computerized medical imaging and graphics*, 27(2-3):185–196, 2003.

[60] NEMA. Digital imaging and communications in medicine (dicom) digital signatures. ftp://medical.nema.org/medical/dicom/final/sup41_ft.pdf, 2001.

[61] A. Ghoneim, G. Muhammad, et al. Medical image forgery detection for smart healthcare. *IEEE Communications Magazine*, 56(4):33–37, 2018.

[62] A. Rössler, D. Cozzolino, et al. Faceforensics++: Learning to detect manipulated facial images. *arXiv preprint arXiv:1901.08971*, 2019.

[63] F. Matern, C. Riess, et al. Exploiting visual artifacts to expose deepfakes and face manipulations. In *2019 IEEE Winter Applications of Computer Vision Workshops (WACVW)*, pp. 83–92. IEEE, 2019.

[64] S. Tariq, S. Lee, et al. Detecting both machine and human created fake face images in the wild. In *Proceedings of the 2nd International Workshop on Multimedia Privacy and Security*, pp. 81–87. ACM, 2018.

[65] D. Cozzolino, J. Thies, et al. Forensictransfer: Weakly-supervised domain adaptation for forgery detection. *arXiv preprint arXiv:1812.02510*, 2018.

[66] L. Zheng, Y. Zhang, et al. A survey on image tampering and its detection in real-world photos. *Journal of Visual Communication and Image Representation*, 58:380–399, 2019.

[67] M. Huh, A. Liu, et al. Fighting fake news: Image splice detection via learned self-consistency. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 101–117. 2018.

[68] D. Cozzolino and L. Verdoliva. Noiseprint: a cnn-based camera model fingerprint. *arXiv preprint arXiv:1808.08396*, 2018.

[69] P. Korus and J. Huang. Multi-scale analysis strategies in prnu-based tampering localization. *IEEE Trans. on Information Forensics & Security*, 2017.

Appendix

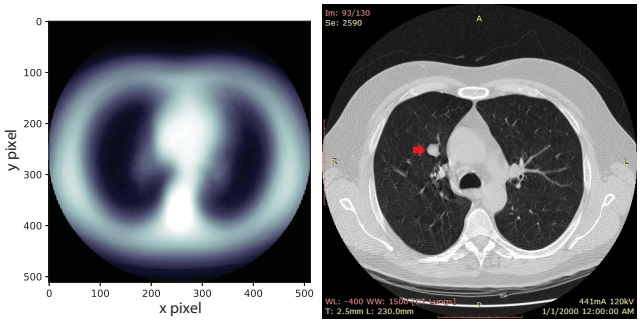

Figure 16: Left: the average of 888 CT scans' middle slices *before* scaling to 1:1:1 ratio (black ares are candidate injection locations). Right: a full slice with an injected nodule.

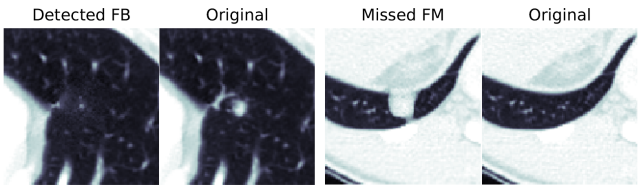

Figure 17: Examples of where the attack failed in the blind trial. Left: a removal (FB) detected as 'ground-glass' cancer due to too much additive noise. Right: an injection missed due to human error.

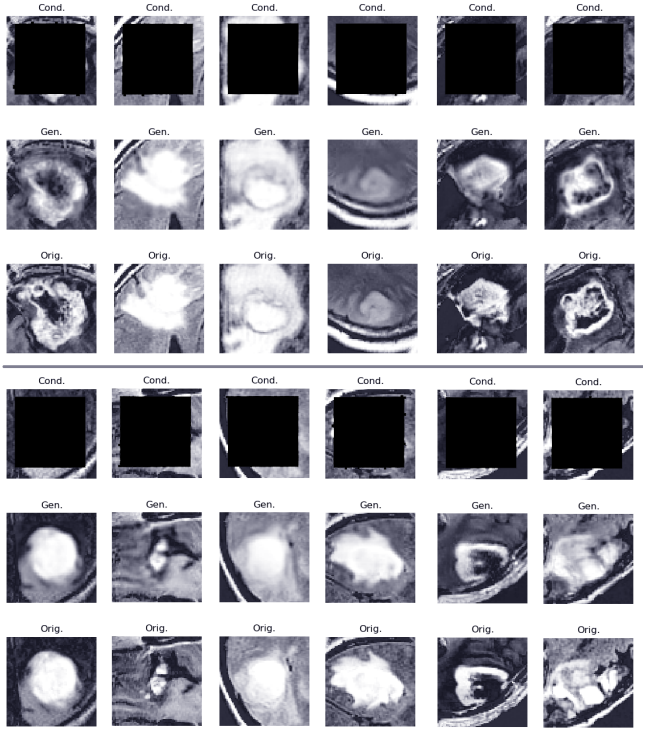

Figure 18: CT-GAN used to inject brain tumors into MRIs of healthy brains. Similar to Fig. 7, Top: context, middle: in-painted result, and bottom: ground-truth. Showing one slice in a 64x64x16 cuboid.

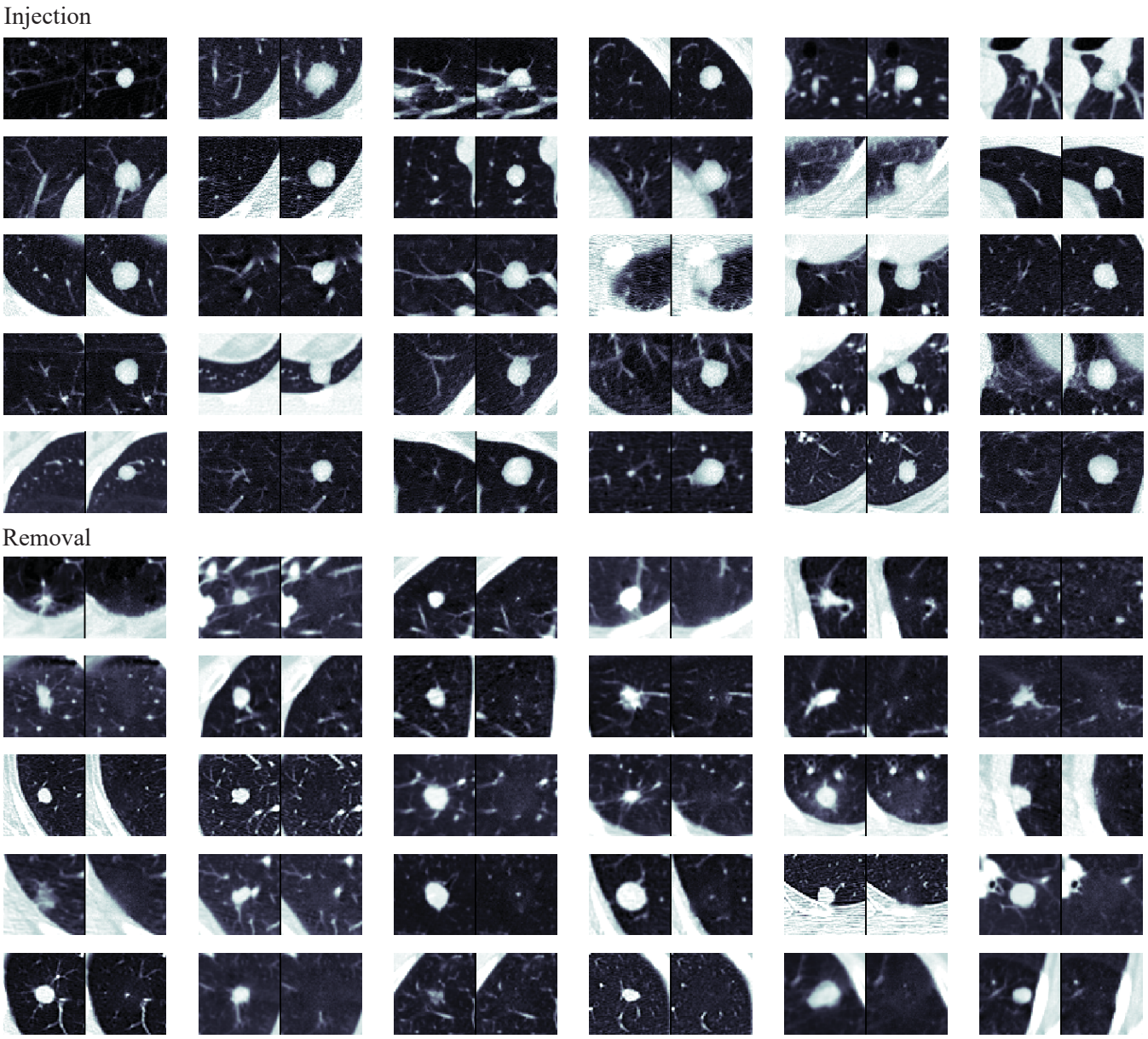

Figure 19: Samples of injected (top) and removed (bottom) pulmonary nodules. For each image, the left side is before tampering and the right side is after. Note, only the middle 2D slice is shown and the images are scaled to different ratios (the source scan).

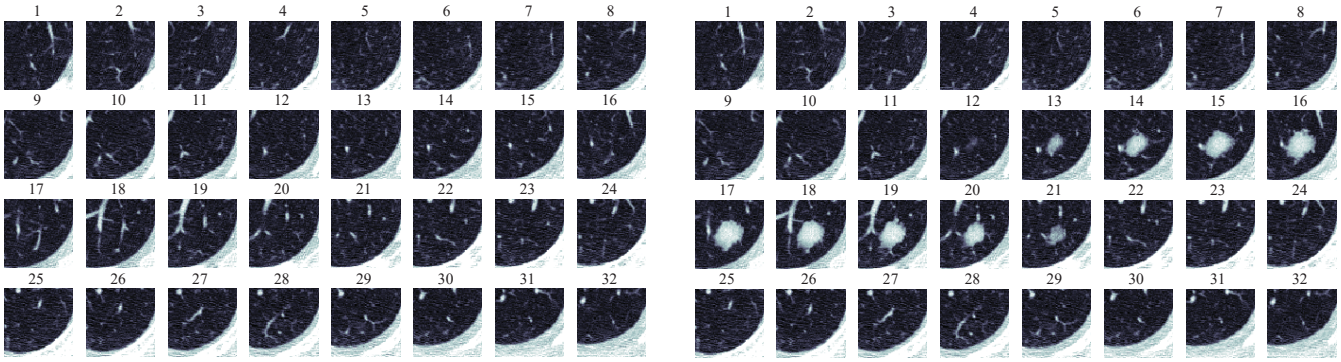

Figure 20: All 32 slices from a sample injection before (left) and after (right) tampering with the CT scan.