Security Framework Implementation Proposal for Teleoperated Surgical Robotics and Medical Devices

Kaitlyn Dean
Département Reseaux et
Télécommunications, Institut
Universitaire de Technologie 1
Université Grenoble Alpes
Grenoble, France
kaitlyndean9@gmail.com

***Abstract*—*In telesurgical environments, surgeons and patients are connected remotely through public networks which gives rise to security and privacy risks. In this synthesis, security measures and vulnerabilities are discussed as well as potential protocols, configurations, and topologies. In health care environments, patient confidentiality and well-being are very critical and must be protected and verified, approaches such as user authentication and data encryption strategies should be utilized in order to guarantee the authenticity of transmitted data.*

Keywords — Cyber physical systems, security, telesurgical robotics, telesurgery, Advanced Encryption Standard, preoperative setup, robotic surgery, key agreements

I. Background

Teleoperated surgery and medical devices have gained in popularity in the past few years. This allows medical procedures to be done remotely which opens a range of possibilities and gives the feasibility of high-end medical access in remote locations such as battlefields. They can be deployed in uncontrolled, adversarial situations or in secure settings, however, protection against compromises and attacks must be prepared. These robots run on a single PC with open-source software such as Linux and Robot Operating Systems (ROS) [1]. Using a standard communication protocol known as the Interoperable Telesurgery Protocol, the robot can communicate to the control console.

After the Health Insurance Portability and Accountability Act (HIPAA) was established, the primary concern became patients' privacy and confidentiality of data storage and transmission. The experts of SecureSurgiNET said "Patients' safety, security, and data privacy are some of the major obstacles and concerns in these types of procedures" [2] and many other teams are working towards a solution to these conditions. Some of these worries concentrated on realization of newly discovered vulnerabilities such as manipulation of medical devices like pacemakers. This motivated research that recently discovered the beneficial use of the Transport Layer Security (TLS).

Teleoperated robots use publicly available networks and temporary ad-hoc wireless and satellite networks to transmit sensory data between controller and remote robotics [3]. This provides immediate medical relief in rural terrains and disaster locations. Lack of protection of these systems may be detrimental and targeted attacks on programmable logic controllers have occurred, Stuxnet worm for a recent example.

The Department of Homeland Security (DHS) has identified telesurgical robots with cyber physical systems as a class where hardware, software and security must develop concurrently. This will push the development of security enhancements in surgical robotics. There are predictions that the development of communication standards for telesurgical robotics systems will progress similarly to connected heterogeneous computer systems which will speed development, research, and deployment of these robotics. Future standards should include session negotiation by implementations such as user exclusion and authentication which will be looked at in this synthesis along with security risk solutions and measures that should be added to a framework standard for surgical robotics and medical devices.

II. Application of Security Policies, Standards, and Protocols

Taking advantage of the already existing Interoperable Telesurgery Protocol (ITP) allows the ability to enhance the security of telesurgical robotics. Utilizing external Open-Source software packages such as OpenSSL can provide secure communication links and tools for incorporating other types of security.

The ITP is a shared data interface intended for the commencement of a standard. There are a wide variety of teleoperated robots differing in scale, application, kinematics, and dynamics which make it hard to create one shared standard. The extreme delay sensitivity and high packet rates make many communication architectures and open-source software unsuitable to telemanipulators and teleoperated surgical systems [4].

A few other unexplored solutions to mention are a medical imaging protocol called OpenIGTLink and a software application known as the Surgical Assistance Workstation.

Fortifying the ITP with the use of certificates provides authentication levels on top of the default addressed authentication element as well as communication, authorization, and security policy development and enforcement. Defining these security policies demonstrate how these concerns need to be acknowledged. Leveraging established security techniques and existing communication protocols specific to telesurgical robotics will accelerate introductory security development. “This protocol ensures privacy and information integrity on individual communication channels” [5]. ITP has begun to cover software, hardware verification and attestation. Furthermore, dedicated point-to-point network links have been used in successful operation in long distance surgery and the communication architecture is often a network socket with a defined data interface to transmit data in fixed binary packets.

Telesurgical operators have a common feature called Scaling for user’s motion, the factor is not explicitly encoded but instead is controlled by the surgeon-side robot and is implicit in the motion commands. Another feature allows controller repositioning without the movement of the patient-side robot called Indexing. This introduces two defined states to coordinate Indexing in “surgeon_mode”: “engaged” where motion commands are followed and “disengaged” where motion commands are ignored [6].

ITP employs two communication channels [5]; between patient-side and surgeon-side telerobotics. The supervisory channel calls upon the Transmission Control Protocol (TCP) to allow the communications to reach the destination with moderate delay tolerance. The commanded inputs from controller to patient-surgery robot and back to the controller harnesses a channel that transmits data every millisecond. It can withhold a minimal quantity of intermittent packet loss but is not as resilient to delays above minor. Therefore, this second communication channel benefits from the User Datagram Protocol (UDP).

A. *Transmission Control Protocol (TCP)*

With TCP, we can use a secure communication protocol referred to as Transport Layer Security (TLS). It uses Advanced Encryption Standard (AES) to encipher to classify protected information with an adequate number of key lengths of 192 and 256 bits. This can safeguard the privacy and integrity of the data. This level of data security commemorates with civil environments and military and can be accomplished using available computing hardware [5].

B. *User Datagram Protocol (UDP)*

With UDP, we can use a secure communication protocol derived from TLS and referred to as Datagram Transport Layer Security (DTLS) [Fig. 1].

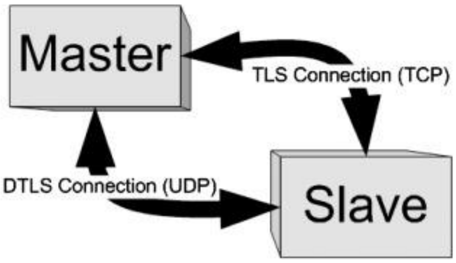
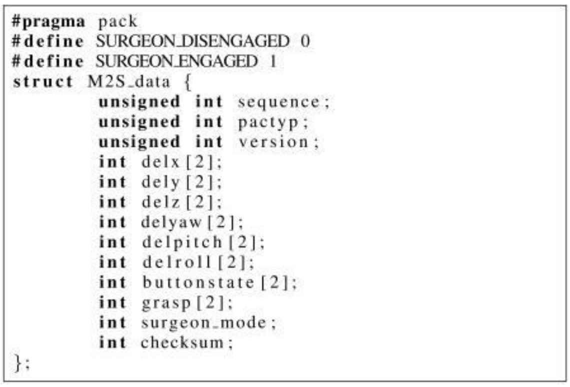


Fig. 1. TCP and UDP communication channels. [1]

C. *Interoperable Telesurgery Protocol (ITP)*

The ITP (a stateless data description) was tested in an experiment where all systems used UDP/IP. The experiment consisted of six slave systems (patient-side robots) [Appendix A] and eight master systems (controllers) [Appendix B] transmitting data up to 1kHz via UDP/IP [7]. Other similar tests have been conducted at high-speed networks operating at 10 gigabits/second, 100 Mb/second minimum bandwidth, 1kHz UDP traffic, and 30fps video feeds with 33ms coding and decoding. This lightweight UDP can disregard corrupt packages and rely on data representation for consistency [8].

Success was measured by the robustness to packet loss and arbitrary network delays, distinctly, recovery from major discontinuities and delay periods. Motions and robotic arm positions are encoded in Cartesian integer dimensions – avoiding floating point numbers, namely delx, dely and delz as well as delroll, delpitch and delyaw [Fig. 2]. See [Fig. 3] for how these positions were calculated. “Endianness and data types were chosen to conform to 32-bit ×86 architecture” [6].



```
#pragma pack
#define SURGEON_DISENGAGED 0
#define SURGEON_ENGAGED 1
struct M2S_data {
        unsigned int sequence;
        unsigned int pactyp;
        unsigned int version;
        int delx[2];
        int dely[2];
        int delz[2];
        int delyaw[2];
        int delpitch[2];
        int delroll[2];
        int buttonstate[2];
        int grasp[2];
        int surgeon_mode;
        int checksum;
};
```

Fig. 2. C Struct UDP packet structure. [6]

$$
\begin{aligned}
\Delta X_k &= X_k - X_{k-1} \\
\Delta R_k &= R_k - R_{k-1}
\end{aligned}
$$

$$
\begin{aligned}
X &= [x, y, z]^T \\
R &= [roll, pitch, yaw]^T
\end{aligned}
$$

Fig. 3. Position formula [6]

The UDP provides minimum possible latency while evading the need of additional overhead like the TCP. A sequence number and checksum are important on packets to safeguard lost, corrupt, and out-of-sequence packets. This will dissuade unpredictable, unsafe patient-side robotic orientations [7].

D. Authentication and Encryption (AES)

The surgeon and patient [Fig. 4] can be authenticated with X.509 certificates as well as medical support staff, instruments and more. Certificates can be verified with parameters such as timestamps and public keys of the certificate owner.

Double encrypting can be done to encrypt the message as well as adding an enciphered signature. This version of the certificate specification features the ability to implement custom fields permitting preliminary authorization levels which currently exist including Maintenance, Non-Surgical, Non-Human, Human, and Override [5] [TABLE I.] [TABLE II.]. The ITP currently enciphers up to a standard specified by the Federal Information Processing Standards (FIPS) by National Institute of Standards and Technology (NIST). TABLE I.

An example certificate for a surgeon could include the surgical procedure and the maximum time delay.

A patient certificate could include surgical procedure, allergies, and even link to the performing surgeons.

These require standard, formal, and systematic protocols [9]. Analysis can be done on the authentication and security in different ways formally and informally using verification tools such as the widely accepted Real-Or-Random (ROR) model with a tool like Automated Validation of Internet Security Protocols and Applications (AVISPA), this tool and model will work for the formal verifications [9]. We can also use protection measures such as semantic data protection and syntactic data protection. We can hide the patient identity with pseudo identity usage.

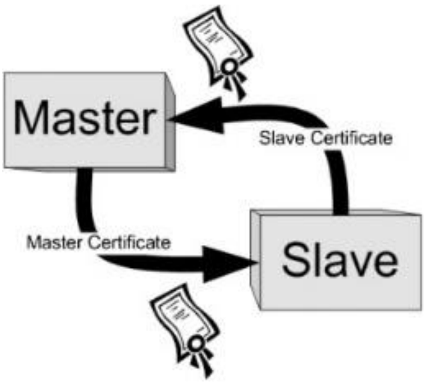
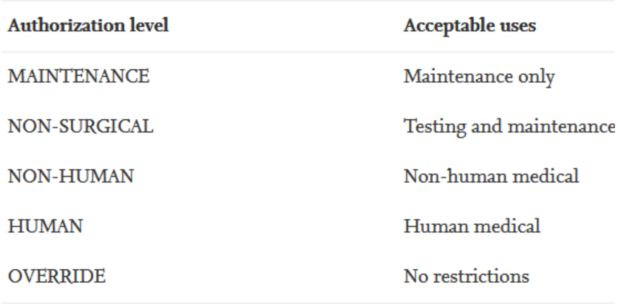


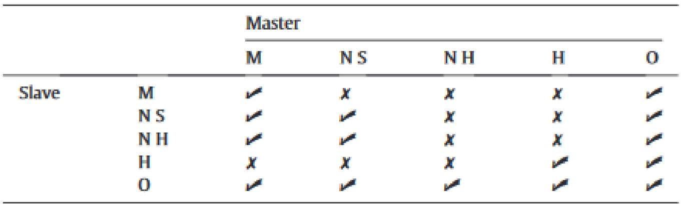
Fig. 4. Certificate authentication between controller and patient robots. [1]

TABLE I. ACCEPTABLE USES CORRESPONDED TO AUTHORIZATION LEVEL. [1]



Authorization level	Acceptable uses
MAINTENANCE	Maintenance only
NON-SURGICAL	Testing and maintenance
NON-HUMAN	Non-human medical
HUMAN	Human medical
OVERRIDE	No restrictions

TABLE II. PRELIMINARY SECURITY POLICIES. [1]



		Master				
		M	N S	N H	H	O
Slave	M	✓	✗	✗	✗	✓
	N S	✓	✓	✗	✗	✓
	N H	✓	✓	✗	✗	✓
	H	✗	✗	✗	✓	✓
	O	✓	✓	✓	✓	✓

III. Risks

Without a set framework standard, the security risks increase significantly. The lives of patients are put on the line with surgical and medical robotics.

With endpoint attacks, the hacker can control and access the robot and data as if they were the surgeon. This imposes a risk to patient confidentiality and the well-being of the patient.

Similarly, with Over-The-Air attacks, the attacker can view the data and attack the movements and behaviour of the robot. This like endpoint attacks, puts emphasis on the need for patient confidentiality protection and data integrity.

Unintended movements of the robot arms can do fatal damage on the patient since the robots are often involved in microscopic surgeries. Medical device manipulation can also be life threatening with unpredictable behaviours; An example of this could be if a pacemaker was altered and acting sporadically.

Adding data integrity, privacy, and other securities into a framework standard optimized and specialized for surgical robotics and medical devices would help to prevent and diminish any risks to the patient and healthcare system.

IV. Attacks

In robot telecommunications, there are two attack vectors, network communication attacks and endpoint attacks. Network communication attacks will likely be more common as endpoint access is more strictly monitored. Attacks resulting in unintended robot movements delays and related unexpected behaviours are known as intention modification and happen while packets are modified in transmission.

Attacks where feedback messages are manipulated, the surgeon's valid actions may become harmful to the patient.

These can be difficult to mount as there is a large amount of data transmitting and these manipulations appear very subtly.

Another attack known as hijacking causes the robot to ignore the surgeon commands and commit potentially harmful movements instead. The attacker may initially eavesdrop on transmissions before injecting, this is known as a network observer. Another method an attacker may take is when they completely stop through-communications and control all robot actions, this is known as a network intermediary and may present itself as ARP Poisoning [1].

An attack in September of 2020 was made against a German Hospital where it resulted in the death of a woman. Other similar attacks against healthcare technology are "WannaCry" and "NotPetya". This shows the importance of having protected devices to ensure the best treatment for all.

A. *Denial of Service (DoS) Attack*

Denial of Service attacks can create threats to patients' well-being. We can prevent this with methods such as incoming connection tracking, reverse proxy, and local wrong input detection.

B. *Man in the Middle Attack*

Injection tests were conducted and succeeded. This happened because valid packets from any source were accepted and solutions need to be made, especially against sophisticated packet spoofing attacks. This can be helped with encryption on all data stream endpoints which eliminate Man in the Middle attacks. By authenticating and encrypting these data points, the intention modification, hijacking, and manipulation become more protected against and hampers the initiation ability of an attacker.

Encrypting costs were investigated by using a Raven robot and an intermediary computer to execute cryptographic tasks – with Intel Core2 Quad CPU processor running at 2.5GHz [10]. Using AES and considering 128-bit, 192-bit, and 256-bit lengths, there was no drastic CPU usage increase but there was an observable memory usage increase with an average of 3000KB [10]. Therefore, this method – encrypting has low cost with high benefits for mitigating these attacks.

C. *False Data Injection*

Some systems such as Raven II run a servo-loop at a rate of 1000Hz with key functions "(i) coordinate transformations, (ii) forward and inverse kinematics, (iii) gravity compensation, and (iv) joint-level closed-loop feedback control" [3]. The Raven II uses eight channels for signal output control for each joining controller on an interface board with each channel using high resolution 16-bit digital-to-analog conversions [3]. This can perform read/write cycles on all channels in 125 microseconds and control both Raven arms with just the one controller. The Raven II was tested in the Mojave Desert utilizing a UAV-enabled wireless network. Using these public networks, it invites easier malicious intents to the communications. Relying on the systems' dynamics such as optimization and a Kalman filter can mitigate and guarantee detection of some attacks like false data injection, integrity attacks, and replay.

V. Operating System

In 2019, Blackberry QNX became compliant with IEC 62304 Class C for Medical Devices [11] which is endorsed by the FDA and by the Directorate-General for Health and Consumer. QNX has the most advanced and secure embedded software platform.

Blackberry says it also has,

> "binary code scanning SAST to provide insight into the quality and security of software components, preventative software maintenance over the lifetime, world-class services including managed PKI authentication, FIPS-certified encryption, secure OS, penetration testing, secure OTA, updates and lifecycle health monitoring, key injection in silicon chips used in ECUs to act as roots of trust, and isolates safety-critical systems from non-safety critical systems" [11].

Unlike a monolithic OS that places critical OS components together, the Blackberry QNX OS separates them into their own protected memory partitions. It is suited for mission-critical systems and prioritizes safety and reliability with mechanisms like adaptive partitioning. It is used in security and defense, robotic systems, and industrial automation. It allows safe operation on the same system-of-a-chip (SoC) and is POSIX-compliant. It also has solutions that use a black channel approach which ensures the safety of communication transmissions solving the concern of altered data. The system helps provision trust anchors, cryptographic keys, debug passwords, and unique device identifiers [11]. Blackberry OS mitigates attack risks through memory separation, process privilege control, run-time isolation, and compute resource isolation.

Blackberry QNX OS provides secure boot, system privilege levels, cryptographic algorithms, secure over-the-air (OTA) software updates, and AES-256 encrypted, self-verifying filesystem [11].

VI. Conclusion

Surgical robotics and medical devices are ground-breaking technology but with these upcoming technologies, there is a high importance on patient confidentiality and health. These robots hold life risking powers and if hijacked or manipulated, a patient's life could be harmed.

With these risks in mind, preventive actions can mitigate these security threats. Using strategies like a Kalman filter, data encryption, data and user authentication, and user integrity can make threat of namely a few major potential attacks such as denial of service, man-in-the-middle, and false data injection attacks. These prevent over-the-air attacks but addressing the end point protection is also important. Blackberry QNX OS provides that end point protection as well as over-the-air measures to cover many different bases of protection. With these implementations into a new secure framework standard for surgical robotics and medical devices, patient privacy and health can be ensured.

References

[1] Emerging Technology from the arXiv, "Security Experts Hack Teleoperated Surgical Robot," MIT Technology Review, p. 2.

[2] S. Iqbal, S. Farooq, A. W. Malik, M. M. Hamayun, K. Shahzad and O. Hasan, "SecureSurgiNET: A framework for," International Journal of Distributed Sensor Networks, p. 12, 2019.

[3] T. Bonaci, J. Herron, T. Yusuf, J. Yan, T. Kohno and H. J. Chizeck, "To Make a Robot Secure: An Experimental Analysis of Cyber Security Threats Against Teleoperated Surgical Robotics," arXiv, p. 11.

[4] H. H. King, B. Hannaford and T. Low, "Plugfest 2009: Global Interoperability in Telerobotics and Telemedicine," IEEE International Conference on Robotics and Automation : ICRA : [proceedings] IEEE International Conference on Robotics and Automation, p. 14.

[5] G. S. Lee and B. Thuraisingham, "Cyberphysical systems security applied to telesurgical robotics," Computer Standards & Interfaces, p. 5, 2011.

[6] H. King, B. Hannaford, K.-W. Kwok, G.-Z. Yang, P. Griffiths, A. Okamura, I. Farkhatdinov, J.-H. Ryu, G. Sankaranarayanan, V. Arikatla, K. Tadano, K. Kawashima, A. Peer, T. Schauss, M. Buss, L. Miller, D. Glozman, J. Rosen and T. Low, "Plugfest 2009: Global Interoperability in Telerobotics and Telemedicine," p. 16, 2014.

[7] H. H. King, B. Hannaford, K.-W. Kwok, G.-Z. Yang, P. Griffiths, A. Okamura, I. Farkhatdinov, J.-H. Ryu, G. Sankaranarayanan, V. Arikatla, K. Tadano, K. Kawashima, A. Peer, T. Schuaß, M. Buss, L. Miller, D. Glozman, J. Rosen and T. Low, "Plugfest 2009: Global Interoperability in Telerobotics and Telemedicine," US National Library of Medicine, National Institutes of Health, p. 14, 2016.

[8] H. H. King, B. Hannaford, K. Tadano, R. Donlin, D., "Preliminary Protocol for Interoperable Telesurgery," p. 6, 2009.

[9] M. Wazid, A. Kumar Das and J.-H. Lee, "User Authentication in a Tactile Internet Based Remote Surgery Environment: Security Issues, Challenges, and Future Research Directions," p. 15, 2019.

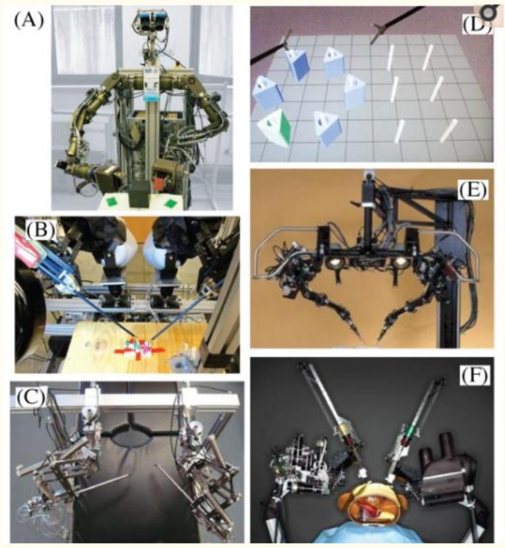
[10] T. Bonaci, J. Herron, T. Yusuf, J. Yan, T. Kohno and H. J. Chizeck, "To Make a Robot Secure: An Experimental Analysis of Cyber Security Threats Against Teleoperated Surgical Robotics," p. 11, 2015.

[11] Blackberry, "Embedded Systems: Safety, Security and Reliability," p. 3, 2021.

APPENDICES

Appendix A *List of Patient-Side Robotics*

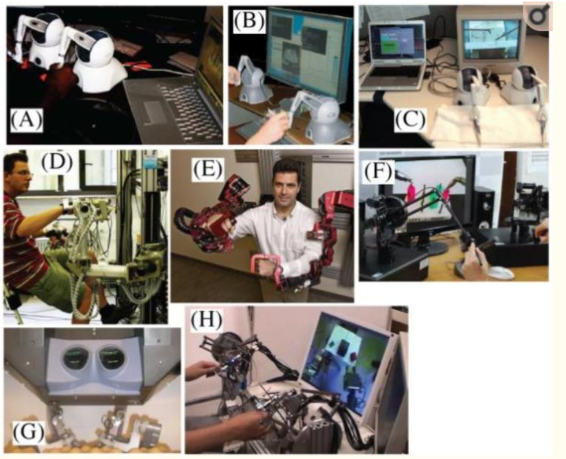
In [3],



(A) TUM general purpose Telerobot.
(B) Patient-side robot of the JHU custom version of the da Vinci.
(C) TokyoTech IBIS IV surgical robot.
(D) RPI VBLaST™. I SRI M7 surgical robot.
(F) UW Raven surgical robot.

Appendix B *List of Surgeon-Side Robotics*

In [3],



(A, B, C) Phantom Omni control station with free software at RPI, ICL and UW respectively. (D) TUM ViSHaRD7. I UCSC Exoskeleton.
(F) Phantom Premium with custom software at KUT.
(G) Master console of the JHU custom version of the daVinci.
(H) TokyoTech delta master.