ML-Driven Quantum Hacking of CHSH-Based QKD

Speaking Notes for 20-Minute Presentation

19 Slides | Target Duration: 19-20 minutes

SLIDE 1: Title Slide

Time: 0:00-0:50 | Duration: 50s

On screen:

- Machine Learning-Driven Quantum Hacking of CHSH-Based QKD
- Subtitle: Exploiting Entropy Vulnerabilities in Self-Testing Protocols
- Authors: Hubert Kolcz^1, Tushar Pandey^2, Yug Shah^3
- ^1Warsaw University of Technology | ^2Texas A&M University | ^3University of Toronto

What to say:

- Good morning. I'm Hubert Kolcz from Warsaw University of Technology
- Today we present ML-driven analysis of entropy vulnerabilities in CHSH-based quantum key distribution
- Key finding: ML can fingerprint quantum RNGs at 59% accuracy even when CHSH tests pass
- Validated on N=30 synthetic devices with 20x distinguishability between classes

SLIDE 2: CHSH Inequality: Foundation for QKD Security

Time: 0:50-2:10 | Duration: 80s

On screen:

- Equation: S = <AB> + <AB'> + <A'B> - <A'B'>
- Classical: S <= 2 | Quantum: S > 2 (max 2sqrt2 ~= 2.828)
- Why CHSH Dominates QKD Industry

What to say:

- CHSH inequality is foundation for device-independent QKD
- Combines four correlation measurements between Alice and Bob
- Classical physics constrains S <= 2, quantum allows up to 2.828
- Why industry adopts CHSH:
- Experimental robustness: tolerates detector imperfections (unlike Bell's perfect correlation requirement)
- Self-testing capability: simultaneously verifies quantum state AND detects eavesdropping
- Device-independent security: if S > 2, eavesdropper cannot have complete key information
- Industry standard in metro QKD networks and commercial implementations

SLIDE 3: The Critical Security Gap

Time: 2:10-3:30 | Duration: 80s

On screen:

- The Paradox: CHSH provides mathematical security, but implementations rely on RNGs susceptible to side-channel attacks
- Four attack types: Phase Remapping, Trojan Horse, Time-Shift, Detector Blinding
- RNG Entropy Analysis box

What to say:

- The central paradox: CHSH gives mathematical security, but implementations depend on vulnerable RNGs
- Traditional attacks target transmission/detection:
- Phase remapping: manipulates quantum phase relationships
- Trojan horse: light signal injection for eavesdropping
- Time-shift: exploits detection timing windows
- Detector blinding: forces detectors into classical mode
- Our work focuses on different vulnerability: RNG entropy source itself
- Our contribution: ML-driven framework analyzing RNG noise through:
- Entropy monitoring (Shannon, min-entropy, KL divergence)
- Hardware metrics (gate fidelity -> Bell correlation)
- CRITICAL DISCLAIMER: We demonstrate statistical fingerprinting on simulator data
- Have NOT demonstrated actual key extraction from real QKD systems
- Gap between detecting patterns and cryptanalysis is significant

SLIDE 4: Multi-Method ML Benchmarking Framework

Time: 3:30-5:00 | Duration: 90s

On screen:

- Three independent approaches: qGAN (12-qubit), Logistic Regression (baseline), Neural Network (30-20-3)
- Four-stage operation flowchart
- Platform comparison: N=3 IBMQ simulators, N=30 synthetic validation

What to say:

- Tested three independent ML approaches for robustness:

1. qGAN: 12-qubit quantum GAN, KL divergence 0.05-0.20

2. Logistic Regression: linear baseline for comparison

3. Neural Network: 30-20-3 architecture, our primary method

- Four-stage operation:

1. Analyze entropy patterns (Markov transitions, autocorrelation, run-length statistics)

2. Correlate hardware metrics (gate fidelity -> Bell correlation, R^2=0.977)

3. Compare platforms (Rigetti 0.8036, IonQ 0.8362, IBM 1.0 baseline)

4. Extract device-specific fingerprints

- Cross-method validation: Pearson r=0.865 (p<10^-^9), Spearman rho=0.931 (p<10^-^1^4)
- All three methods converge on same device classifications

SLIDE 5: Experimental Methodology & Hardware

Time: 5:00-6:30 | Duration: 90s

On screen:

- Hardware platforms table: Rigetti Aspen-M-3 (80 qubits), IonQ Aria-1 (25 qubits), IBM Qiskit (ideal)
- Dataset transparency: N=3 from IBMQ simulators, N=30 synthetic
- Multi-method benchmarking approach

What to say:

- Hardware platforms:
- Rigetti Aspen-M-3: 80 superconducting qubits, Bell correlation 0.8036, gate fidelity 93.6%
- IonQ Aria-1: 25 trapped ion qubits, Bell 0.8362, gate fidelity 99.4%
- IBM Qiskit: ideal simulator baseline, Bell 1.0, fidelity 100%
- Dataset transparency - this is critical:
- N=3 dataset: 6,000 samples from IBMQ noise-injected simulators (DoraHacks YQuantum 2024)
- NOT actual QPU data - realistic noise models but still simulators
- N=30 dataset: entirely synthetic devices with controlled bias levels
- Why synthetic? Proper statistical power requires 30+ samples, commercial QPU access expensive/limited
- N=3 device fingerprints:
- Device 0: 54.7% '1' frequency, P(1->1)=0.573
- Device 1: 56.5%, P(1->1)=0.592
- Device 2: 49.2%, P(1->1)=0.508
- Cross-method validation: r=0.865, rho=0.931, both p<10^-^9

SLIDE 6: Quantitative Analysis: Bit Frequency Distribution

Time: 6:30-7:30 | Duration: 60s

On screen:

- Figure showing bit frequency histograms for 3 devices
- Three colored boxes showing Device 1, 2, 3 statistics

What to say:

- The entropy paradox - this is fascinating:
- Device 1: 54.8% '1' bias, entropy 0.986 bits (low bias)
- Device 2: 56.5% bias, entropy 0.979 bits (medium bias)
- Device 3: 59.2% bias, entropy 0.992 bits (HIGHEST bias, HIGHEST entropy!)
- Why paradoxical?
- Device 3 has strongest frequency bias but highest Shannon entropy
- Yet easiest to classify at 70% accuracy
- Shannon entropy measures first-order statistics (individual bits)
- ML detects second-order patterns (Markov transitions, autocorrelation, run-length)
- Security implication: Passing NIST chi-square tests doesn't guarantee ML-robustness
- All three devices pass chi^2 < 3.841, yet distinguishable at 59-70% accuracy

SLIDE 7: Markov Chain Transition Matrices

Time: 7:30-8:20 | Duration: 50s

On screen:

- Figure showing 3 Markov transition matrix heatmaps
- Device-specific transition probabilities

What to say:

- Markov transition probabilities reveal device fingerprints:
- Device 1: P(1->1)=0.573 (moderate persistence)
- Device 2: P(1->1)=0.592 (strongest persistence - bit '1' tends to repeat)
- Device 3: P(1->1)=0.508 (most symmetric, appears random)
- Key finding: Device-specific transition biases create ML fingerprints invisible to NIST tests
- Memory attack connection:
- P(1->1) deviation from 0.5 indicates statistical memory
- Physical origins: detector afterpulsing, gate errors, thermal drift
- Enables basis prediction with better-than-random probability
- Standard NIST tests check frequency distributions, not temporal correlations

SLIDE 8: Machine Learning Performance Metrics

Time: 8:20-9:50 | Duration: 90s

On screen:

- Four-panel figure: confusion matrix, method comparison, N=3 to N=30 bridging, baseline improvement

What to say:

- N=30 Validation Results - four panels:

Panel A - Confusion Matrix:

- 59% overall accuracy balanced across 3 device classes
- Diagonal shows correct classifications, off-diagonal shows confusion

Panel B - Method Comparison:

- Neural Network: 59% accuracy
- Logistic Regression: 60% (slightly better, simpler model)
- Random baseline: 33.3% (3-way classification)
- 77% improvement over random guessing

Panel C - N=3 to N=30 Bridging (critical validation):

- N=3: 58.67% accuracy (IBMQ simulators)
- N=30: 59.21% accuracy (synthetic)
- Replicates within 0.54 percentage points
- p<10^-^9 with 28 degrees of freedom
- Model trained on N=3 generalizes to N=30 without modification

Panel D - Baseline Improvement:

- All methods 77-80% above random
- Statistical significance validated
- Conclusion: Method works on synthetic data with proper statistical power
- Need validation on 50+ real production QKD RNGs

SLIDE 9: Neural Network Architecture Analysis

Time: 9:50-10:40 | Duration: 50s

On screen:

- Architecture diagram or hyperparameter analysis figure

What to say:

- Optimal configuration found through grid search:
- Batch size: 8 (stable gradients without overfitting)
- Architecture: 30-20-3 (wider first layer captures more features, bottleneck for classification)
- Regularization: L1 λ=0.002 (sparse feature selection, prevents overfitting)
- Training: 1000 epochs necessary for convergence
- Generalization evidence:
- Hyperparameters optimized on N=3 dataset
- Applied to N=30 without any modification
- Performance replicates (58.67% -> 59.21%)
- Indicates not overfitting to specific N=3 noise profiles
- Model learned generalizable RNG fingerprints

SLIDE 10: Per-Device Classification Performance

Time: 10:40-11:30 | Duration: 50s

On screen:

- Bar chart or table showing individual device accuracy

What to say:

- Individual device results (N=3):
- Device 1 (54.8% '1' bias): 66.7% classification accuracy
- Device 2 (56.5% bias): 65.0% accuracy - most challenging because similar to both extremes
- Device 3 (59.2% bias): 70.0% accuracy - best performance despite highest entropy!
- Key insight: Device 3 paradox again
- Highest Shannon entropy (0.992 bits)
- Should be "most random" by traditional metrics
- Yet easiest to classify because ML exploits temporal patterns
- Security takeaway: High entropy is necessary but NOT sufficient
- Need adversarial robustness testing beyond NIST
- Second-order statistics matter for ML security

SLIDE 11: qGAN Distributional Analysis: Device Distinguishability

Time: 11:30-12:50 | Duration: 80s

On screen:

- KL divergence tournament results
- Within-class vs between-class comparison

What to say:

- qGAN tournament methodology:
- 435 pairwise KL divergence comparisons
- All combinations of 30 devices in N=30 validation
- Within-class KL (similar devices in same bias class):
- Class 0-0 (low bias): mean 0.048, std 0.044 (very similar)
- Class 1-1 (medium): mean 0.083, std 0.072
- Class 2-2 (high): mean 0.101, std 0.101 (most variable)
- Between-class KL (different bias classes):
- Classes 0-1: mean 0.670, std 0.369
- Classes 0-2: mean 3.180, std 1.318 (MOST distinguishable)
- Classes 1-2: mean 0.961, std 0.705
- Statistical validation:
- Mann-Whitney U test: p=3.26x10^-^6^0 (ridiculously significant)
- 20x higher distinguishability between classes vs within classes
- Confirms devices cluster by bias level
- Cross-method consistency:
- qGAN KL divergence correlates with NN accuracy: r=0.865, rho=0.931
- Independent methods converge on same device structure

SLIDE 12: Proposed DI-QKD Vulnerability Analysis

Time: 12:50-14:20 | Duration: 90s

On screen:

- Two-phase attack methodology (Phase 1: RNG Profiling, Phase 2: Basis Prediction)
- Validated technical foundation section
- Critical finding box

What to say:

- Phase 1: RNG Profiling (passive monitoring)
- Collect RNG output during normal QKD operation
- ML fingerprinting: classify device at 59% accuracy (80% above random)
- Bias detection: identify exploitable 59% vs 54% '1' frequency threshold
- Temporal pattern extraction: Markov transitions P(1->1) = 0.508-0.592
- Phase 2: Measurement Basis Prediction (active exploitation)
- Monitor environmental factors: temperature, gate fidelity drift
- Track CHSH degradation: deviation from ideal 2.828 to exploitable S<2.2
- Basis inference: use RNG bias patterns to predict Alice/Bob measurement settings
- Side-channel extraction: combine entropy deviation + hardware signatures
- Validated technical foundation:
- Multi-modal validation: qGAN-NN correlation r=0.865 (N=30, p<10^-^9), between-class KL 20x higher (p<10^-^6^0)
- Hardware correlation: Gate fidelity -> CHSH score -> RNG quality (R^2=0.977 across Rigetti/IonQ/IBM)
- Attack detection threshold: CHSH<2.2 + bias>59% signals exploitable conditions
- DI-QKD vulnerability context:
- Security proofs assume stable min-entropy bounds
- Don't account for temporal correlations beyond what min-entropy captures
- Our framework detects when real RNGs violate theoretical assumptions
- CRITICAL GAP: We fingerprint certified QRNGs but haven't demonstrated key extraction on real QKD
- Detecting statistical patterns != extracting secret keys
- Real attack requires real-time basis prediction + key correlation
- Gap between our work and actual security breach is substantial

SLIDE 13: Hardware Platform CHSH Correlation Analysis

Time: 14:20-15:20 | Duration: 60s

On screen:

- Table comparing IBM Qiskit, Rigetti Aspen-M-3, IonQ Aria-1
- Hardware comparison figure

What to say:

- Platform comparison table:
- IBM Qiskit (simulator): Bell 1.0, gate fidelity 100%, ideal baseline
- Rigetti Aspen-M-3: 80 superconducting qubits, Bell 0.8036, fidelity 93.6%
- IonQ Aria-1: 25 trapped ion qubits, Bell 0.8362, fidelity 99.4% (best hardware)
- Critical finding: R^2=0.977 correlation
- Gate fidelity strongly predicts Bell correlation coefficient
- Lower gate fidelity -> lower Bell correlation -> noisier quantum operations -> more exploitable RNG
- Easier to measure gate fidelity than full CHSH tests
- Provides early warning system for RNG vulnerability
- Aggregate vs microstructure:
- CHSH verifies aggregate S values (averaged over 100,000 shots)
- Doesn't verify per-sample microstructure (temporal correlations within 100-bit sequences)
- CHSH says "average behavior is quantum"
- ML says "individual patterns are exploitable"
- Gap between aggregate certification and sample-level security

SLIDE 14: Statistical Significance & Correlation Analysis

Time: 15:20-16:10 | Duration: 50s

On screen:

- Correlation analysis figure with scatter plot and confidence intervals
- Two-column boxes: Correlation Evidence, Statistical Power

What to say:

- Correlation evidence:
- Pearson r = 0.865 (p<10^-^9) - strong linear correlation
- Spearman rho = 0.931 (p<10^-^1^4) - even stronger, non-parametric, robust to outliers
- 95% confidence intervals shown in figure
- Homoscedastic residuals (constant variance)
- Statistical power:
- N=30 devices gives df=28 degrees of freedom
- Adequate for detecting medium-to-large effects
- All comparisons p < 0.01
- Mann-Whitney U: p<10^-^6^0 for between-class vs within-class
- 20x higher distinguishability between classes
- NIST test paradox:
- All devices pass chi^2 test (values < 3.841 threshold)
- All have high Shannon entropy (0.979-0.992 bits, near ideal 1.0)
- Yet NN classifies at 59% with p<10^-^9
- Key message: Passing NIST tests is insufficient for ML-adversarial security
- Need second-order statistics evaluation
- Temporal pattern analysis beyond frequency tests

SLIDE 15: Proposed Attack Detection Framework

Time: 16:10-17:20 | Duration: 70s

On screen:

- Two-column comparison: High-Quality RNG vs Degraded RNG
- Attack Type Signatures (4 cards)
- Proposed application note

What to say:

- High-quality RNG profile (safe operation):
- Bell correlation >= 0.8 (high quantum fidelity)
- Shannon entropy ~= 0.99 bits (near ideal)
- KL divergence ~= 3.7 (stable baseline distribution)
- Bit frequency: 50% +/- 2% (within tolerances)
- Degraded RNG profile (attack signatures):
- Bell correlation < 0.8 (increasing noise)
- Entropy deviation > 5% from baseline
- KL divergence > 17 (massive distribution shift)
- Bit frequency: 59% (exploitable threshold from our analysis)
- Attack type signatures (4 categories):
- Phase remapping: correlation drop + entropy oscillation pattern
- Detector blinding: complete loss of quantum correlation (S -> 2)
- Temperature attack: gradual bias accumulation over time
- RNG compromise: persistent frequency bias + Markov patterns
- Proposed application: Real-time statistical monitoring framework
- Multi-indicator thresholds for early detection
- Combines entropy + correlation + hardware metrics
- Critical requirement: Validation on 50+ production QKD RNGs needed
- Current results on synthetic data only
- Need long-term studies on real commercial systems

SLIDE 16: Proposed Application: Metro QKD Security Monitoring

Time: 17:20-18:10 | Duration: 50s

On screen:

- Validated Methods section
- Two metric boxes: 59% accuracy, 20x distinguishability
- Critical gaps note (highlighted in red)

What to say:

- What's validated (on synthetic data):
- Framework tested on N=30 synthetic devices
- 59% classification accuracy, 80% above random, p<10^-^9
- 20x distinguishability between device classes, p<10^-^6^0
- qGAN tournament confirms device clustering
- Statistical signatures detectable despite all devices passing NIST tests
- Critical gaps for real-world deployment:

1. Need 50+ real production QKD RNGs (not synthetic simulators)
2. Long-term drift monitoring: months to years of continuous operation data
3. Demonstration of actual key leakage detection (not just statistical patterns)

- Honest assessment - and this is crucial:
- We detect patterns in RNG output
- Have NOT demonstrated key extraction in practice
- Gap between statistical fingerprinting and cryptanalysis is substantial
- Would need: real-time basis prediction, correlation with key bits, information leakage quantification
- Our work: first step identifying vulnerability, not complete attack

SLIDE 17: Bridging Theory & Engineering Reality

Time: 18:10-19:10 | Duration: 60s

On screen:

- The Fundamental Gap box
- Two columns: This Work Addresses, Future Directions

What to say:

- The fundamental gap:
- Mathematical excellence: CHSH-based QKD provides device-independent security guarantees with rigorous proofs
- Engineering compromise: Real-world implementations rely on RNGs vulnerable to side-channel attacks
- Our solution: Combines CHSH self-testing with ML-driven entropy monitoring to close this gap
- This work addresses:
- Continuous RNG validation (not one-time certification at deployment)
- Environmental factor monitoring (temperature, electromagnetic interference)
- Hardware drift detection (gradual degradation over lifetime)
- Real-time attack identification (before security breach)
- Future directions:
- Photonic & topological qubits: different noise models, new challenges
- Long-term degradation studies: 6-12 month continuous monitoring campaigns
- Quantum ML for detection: quantum algorithms for RNG quality assessment
- NIST/ISO standards development: integrating ML-based monitoring into certification frameworks
- This bridges academic security proofs with operational security requirements

SLIDE 18: Comprehensive Validation Summary

Time: 19:10-19:40 | Duration: 30s

On screen:

- Six-panel comprehensive figure showing all main results

What to say:

- Point to comprehensive figure (6 panels):
- Top left: bit frequency distributions across 3 devices
- Top right: Markov transition matrices showing device-specific patterns
- Middle left: qGAN tournament - 20x distinguishability between classes
- Middle right: NN confusion matrix - 59% balanced accuracy
- Bottom left: hardware correlation - R^2=0.977 gate fidelity -> Bell correlation
- Bottom right: per-device performance - Device 3 has 70% accuracy despite highest entropy
- Summary message: Multi-method consistency, proper statistical power, N=3 to N=30 replication validates approach on synthetic data

SLIDE 19: Conclusions & Impact

Time: 19:40-21:00 | Duration: 80s

On screen:

- Five key contributions (numbered list)
- Impact statement
- Critical gap reminder (red box)
- Acknowledgments, references, contact

What to say:

- Five key contributions:

1. Device fingerprinting: 59% accuracy on N=30 devices (73.2% balanced), 120% above baseline, r=0.865, p<10^-^9

2. Multi-method consistency: qGAN, Logistic Regression, and Neural Network converge, rho=0.931, p<10^-^1^4

3. qGAN tournament: 20x higher distinguishability between classes (p<10^-^6^0), within-class KL 0.077 vs between-class 1.60

4. Scalability: N=3 (58.67%) replicates to N=30 (59%) with strong statistical significance

5. Proposed application: ML-based continuous monitoring framework validated on synthetic data, requires real hardware testing

- Impact statement:
- ML-based fingerprinting successfully distinguishes quantum noise profiles on synthetic data
- Demonstrates vulnerability class that CHSH certification alone doesn't detect
- Proposes operational security layer beyond theoretical proofs
- Critical gap reminder (emphasize this):
- Detecting patterns != exploiting for QKD attacks
- We show RNGs have ML-detectable fingerprints
- Have NOT demonstrated key leakage in production systems
- Real security impact requires actual key extraction demonstration
- Our work: vulnerability identification, not complete exploit
- References:
- Zapatero et al. 2023 (Nature npj Quantum Information) - DI-QKD security analysis
- DoraHacks YQuantum 2024 - quantum randomness generation challenge
- github.com/hubertkolcz/NoiseVsRandomness - open-source repository
- Acknowledgments: Warsaw University of Technology, Texas A&M University, University of Toronto, DoraHacks
- Contact: hubert.kolcz.dokt@pw.edu.pl

Thank you! Questions?