

# マルチキャストにおけるパケットロスしたシーケンス番号の送信の待機時間をもちいたユニキャストでの再送信による再送時間の削減

山崎 雅也<sup>1</sup> 越後谷 滯<sup>1</sup> 串田 高幸<sup>1</sup>

**概要:** IoT 機器のファームウェアアップデートはゼロデイ攻撃を例とするサイバー攻撃よりも早くアップデートを行わなければならない。多数の IoT 機器のファームウェアアップデートに、マルチキャストを使用することで、更新にかかる時間が短くなる。マルチキャストで送信を行うと人や壁といった障害物でパケットロスすることがあり、再送信をしなければならない。課題は、マルチキャストの送信によって生じたパケットロスの再送信が送信時間の増加になることである。提案は、クライアントからサーバにパケットロスしたシーケンス番号をユニキャストで送信していない時間を利用し、サーバからクライアントにユニキャストで再送信する方法を定めた。評価実験はファームウェアに見立てた 100KB のファイルを送信した後にマルチキャストのみで再送信した時と提案方式を使用した時の 2 種類を比較した。この操作を各 10 回行った。またクライアントの ESP32 は 3 台を使用する。評価はパケットを送信した後にマルチキャストのみで再送信した時と提案方式を使用した時の伝送時間を比較した。結果として、提案方式を使用した時の伝送時間はパケットを送信をした後に、マルチキャストのみで再送信した時と比べて平均して約 123 秒から約 92 秒の約 31 秒減少し、約 25.22%の減少率であった。

## 1. はじめに

### 背景

Internet of Things(IoT) とは、インターネットに接続することができる機器のことである [1]。IoT 機器の運用にはファームウェアが必要である [2]。ファームウェアは、IoT 機器と通信を行うソフトウェアの 1 種である [3]。ファームウェアには、バグによるセキュリティの脆弱性がある [4]。したがって、IoT 機器はセキュリティの観点からファームウェアアップデートが必要になる [5]。また、ゼロデイ攻撃よりも早くアップデートを完了させなければならない [6]。複数の IoT 機器へのアップデートを行う場合、マルチキャストを使用することでアップデートにかかる時間を短縮できる [7]。マルチキャストは、同一ネットワーク内の複数のクライアントに対し、パケットを同時に送信する [8]。したがって、マルチキャストによる複数の IoT 機器への同時アップデートを行うことができる [9]。

マルチキャストには、User Datagram Protocol(UDP) という通信プロトコルが使用されている [10]。UDP はコネクションレスのプロトコルである [11]。したがって、UDP は Transmission Control Protocol(TCP) に比べて速くデータを送信できる [12,13]。しかし、UDP は人や壁を例とする

障害物により、パケットロスをする時がある [14,15]。この時、ファームウェアがパケットロスをする時、脆弱性の原因となるバグが修正されない [16]。したがって、ファームウェアアップデートの時にパケットロスがあるならば再送信する必要がある [17]。

パケットを識別する方法にシーケンス番号がある。シーケンス番号により、パケットロスを検知することができる [18]。

スマートホームには、IoT 機器が使用されている [19]。スマートホームは、ネットワークを介して家電の制御と管理を提供するプラットフォームである [20]。例としては、モバイル IP ベースのウェブカメラを通じて、自宅のどこからでも侵入を監視できるものがある [21]。スマートホームに使用されている IoT 機器の中に、IP カメラがある [22]。IP カメラは、カメラとネットワークビデオ技術を組み合わせたビデオ監視システムである [23]。

IoT 機器の 1 つに ESP32 がある。ESP32 とは、低コストであり、Wireless Fidelity (Wi-Fi) および Bluetooth 機能を備えているマイクロコントローラである [24]。Wi-Fi は IoT 機器がインターネット接続を確立できるようにする無線通信技術である [25]。また、Wi-Fi はアクセスポイントから最大 30m の通信を可能とする [26]。

<sup>1</sup> 東京工科大学コンピュータサイエンス学部  
〒192-0982 東京都八王子市片倉町 1404-1

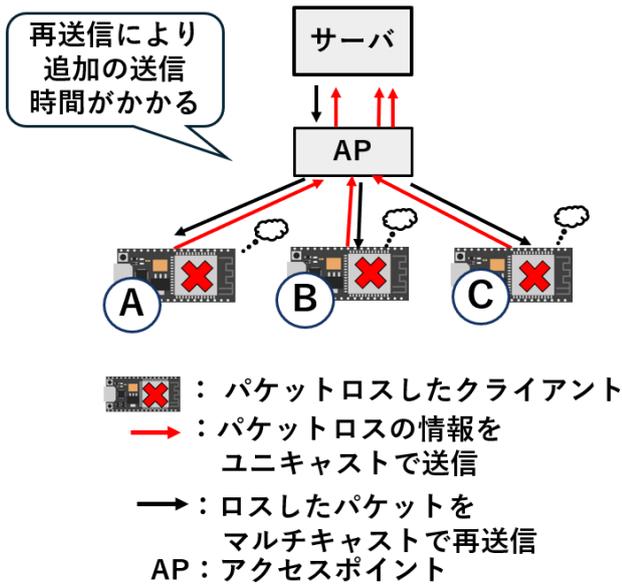


図 1 再送信をしている様子

## 課題

課題は、マルチキャストの送信によって生じたパケットロスの再送信が送信時間の増加につながるである。送信時間の増加によって、ファームウェアアップデートが遅れる原因となる。図 1 は、再送信をしている様子である。人や壁といった障害物によりパケットロスが発生した時、サーバはクライアントにロスしたパケットを再送信しなければならない。この処理により、追加の送信が発生し、本来の送信よりも長くなってしまふ。

## 各章の概要

第 2 章では、本稿の関連研究について述べる。第 3 章では、本稿の課題について解決するための提案方式について述べる。第 4 章では、提案した手法の実装について述べる。第 5 章では、基礎実験として実験内容と実験結果と分析について述べる。第 6 章では、評価実験として実験内容と実験結果と分析について述べる。第 7 章では、提案手法についての議論を述べる。最後に、第 8 章にて結論を述べる。

## 2. 関連研究

Instantly Decodable Network Coding(IDNC) にマルチキャストとユニキャストを適用させることで再送効率を向上させる研究がある [27]。これは、IDNC ベースの方式の Minimum Completion Delay (MCD) 問題をビンパッキング問題としてモデル化をし、有向グラフモデルにもとづいて、すべての Network Coding(NC) 構造を分析した。これらの NC 構造の組み合わせを最適化することで、インターセッションベースのシナリオにおける IDNC の利点の下限を示し、上限を考慮して、多項式複雑度で MCD 問題を解決するための greedy strategy アルゴリズムを提案してい

る。しかし、遅延をどのくらい削減しているかまでは記述がされていなかった。

マルチキャストとユニキャストを組み合わせ、再送回数を減らすネットワークコーディングベースの再送方式を提案している研究がある。これはマルチキャストで発生したパケットロスをマルチキャストで再送信する方法とユニキャストで再送信する方法に分け、再送回数を減らしている。しかし、具体的にどのくらい伝送時間が減少しているのかは明記されていなかった [28]。

クライアントのチャンネルの品質が悪い時、マルチキャストチャンネルの伝送速度がそのクライアントによって遅くなる問題を解決している研究がある。これは事前に定義されたしきい値より低い Signal-to-Noise Ratios (SNR) を持つクライアントに対して、マルチキャストチャンネルに加えてユニキャストチャンネルを割り当てることで、マルチキャストチャンネルの伝送速度を向上させている。しかし、これは再送信時について記述されていなかった [29]。

## 3. 提案

### 提案方式

本稿では、再送信によって生じる時間を削減することが目的である。そこで、クライアントからサーバに、パケットロスしたシーケンス番号をユニキャストで送信していない時間を利用し、サーバからクライアントにユニキャストで再送信する手法を提案する。

図 2 は提案方式の流れである。サーバはクライアントにマルチキャストでパケットを送信している状況である。人や壁といった障害物でパケットロスをした時、クライアントはサーバにマルチキャストでパケットロスしたシーケンス番号をユニキャストで送信する。サーバはクライアントから受け取ったパケットロスしたシーケンス番号から、クライアントにユニキャストでロスしたパケットを再送信する。

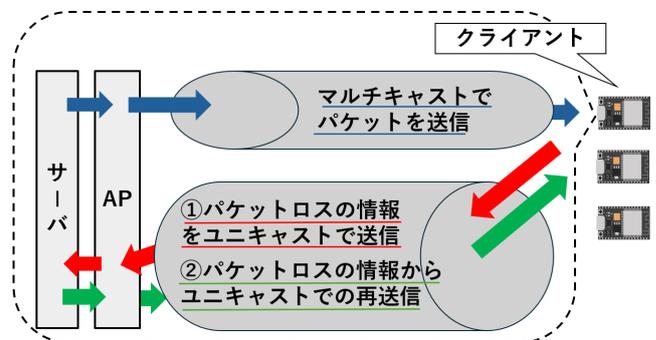


図 2 提案方式の流れ

### ユースケース・シナリオ

本稿のユースケースとして、ホテルでの複数の IP カメ

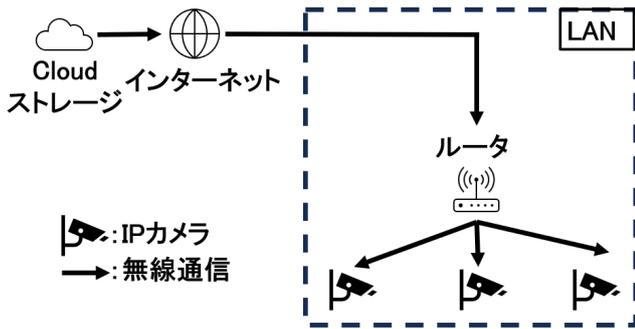


図 3 ユースケースの図

ラの運用を想定している [30]. IP カメラはインターネット経由で画像や動画を送信する機能を備えたカメラである [31, 32]. この際, セキュリティの脆弱性の改善のためファームウェアアップデートが必要になる [33].

図 3 はユースケースの図である. IP カメラはルータを介し, インターネットと繋がり Cloud ストレージの監視情報をやり取りしている [34]. 今回はこのトポロジーを想定する. このユースケースに本提案を使用することで, 送信時間が短縮される.

## 4. 実装

実装では, 提案したアルゴリズムの詳細を説明する. send フェーズと resend フェーズの順で説明する. またクライアントは MicroPython を使用し, サーバは Python を使用して実装を行う.

### 4.1 send フェーズ

図 5 は send フェーズについてである. クライアントとサーバは事前にマルチキャストグループに参加する. サー

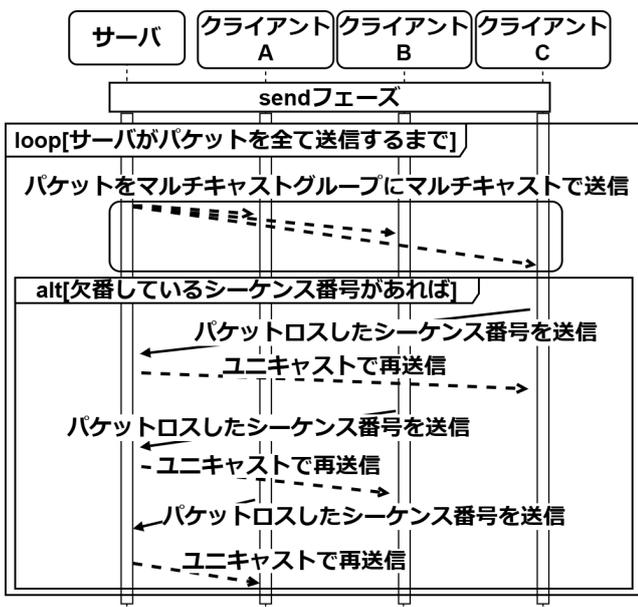


図 4 send フェーズについて

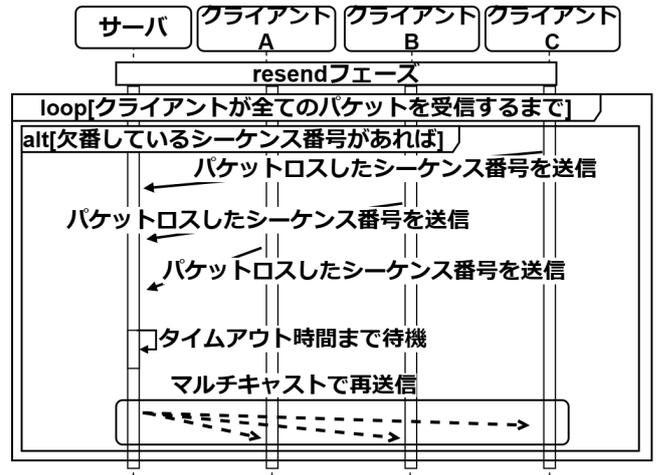


図 5 resend フェーズについて

バからクライアントにマルチキャストでパケットを送信する. 送信するパケットには「シーケンス番号」と「送信したいデータ」がある.

### 4.2 resend フェーズ

図 5 は resend フェーズについてである. クライアントは send フェーズで受け取れなかったパケットロスしたシーケンス番号をサーバにユニキャストで送信する. サーバはそのシーケンス番号のパケットをマルチキャストで再送信する.

## 5. 基礎実験

基礎実験ではパケットロスがあるかを確認した. 実験方法は Access Point (AP) から約 30m の距離に 3 台のクライアントを配置し, 再送信無しのマルチキャストでファイルを送信した. 基礎実験の評価はパケットロス率である. パケットロス率は式 (1) で算出する. PLPer はパケットロス率である. PLNum はクライアントがパケットロスした個数である. TotalNum は送信するパケットの個数である.

$$PLPer = \frac{PLNum}{TotalNum} \times 100 \quad (1)$$

### 実験環境

実験で使用するクライアントは Espressif Systems 社の IoT 機器である ESP32 を使用した. サーバは Broadcom 社の VMware 製品である ESXi を使用した. 図 6 は実験環境を示す. ESP32 を AP から約 30 m の位置に配置した [26]. また ESP32 は 3 台使用した. サーバから送るファイルはファームウェアに見立てた 100KB のファイルを使用した [35]. またパケットサイズは小さい程, 送信効率が良いため 115bytes ごと, 計 891 個のパケットに分割し, 送信した [35, 36]. この操作を 50 回行った.

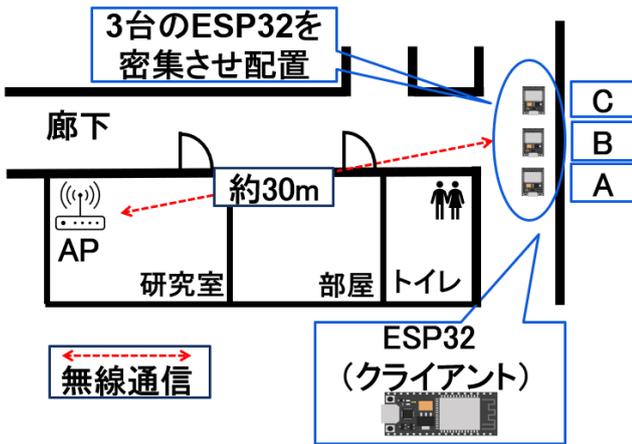


図 6 実験環境について

### 実験結果と分析

図 7 はパケットロス率の結果である。全てのクライアントでパケットロスが確認出来た。よって再送信する必要がある。

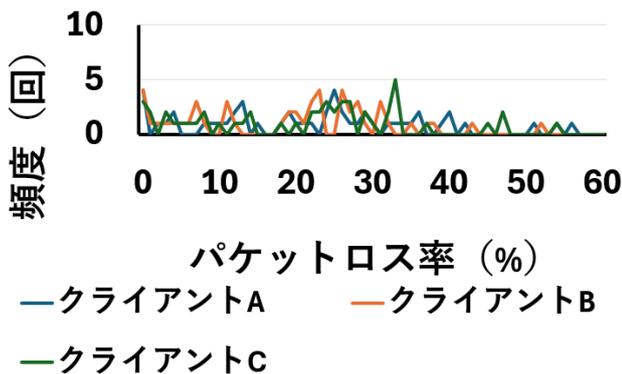


図 7 基礎実験の結果

## 6. 評価実験

評価方法は、パケットを送信した後にマルチキャストのみで再送信した時と提案方式を適用した時の伝送時間を比較する。伝送時間はクライアントがパケットを受信し始めてから全てのパケットを受信し終えるまでの時間とする。

### 実験環境

実験環境は基礎実験と同じ方法で行った。ESP32 は図 6 の配置の方法で行った。また送信するデータはファームウェアに見立てた 100KB のファイルである。このファイルを 115bytes ごと、計 891 個のパケットに分割し、送信した。この操作を各 10 回行った。IoT 機器として Espressif Systems 社の IoT 機器である ESP32 を使用した。サーバとして Broadcom 社の VMware 製品である ESXi を使用した。

### 実験結果と分析

図 8 はパケットを送信した後にマルチキャストのみで再送信した時の伝送時間である。パケットを送信した後にマルチキャストのみで再送信した時の伝送時間の中央値はクライアント A が約 118 秒、クライアント B が約 126 秒、クライアント C が約 127 秒であった。

図 9 は提案方式を適用させた時の伝送時間である。提案方式を適用させた時の伝送時間の中央値はクライアント A が約 94 秒、クライアント B が約 92 秒、クライアント C が約 91 秒であった。

これらの結果より提案方式を使用した時の伝送時間はパケットを送信した後にマルチキャストのみで再送信した時と比べて、クライアント A で約 24 秒減少、クライアント B で 34 秒減少、クライアント C で 36 秒減少した。減少率は、クライアント A で約 20.34%、クライアント B で約 26.98%、クライアント C で約 28.35%であり、平均で約 25.22%減少した



図 8 パケットを送信した後にマルチキャストのみで再送信した時の伝送時間

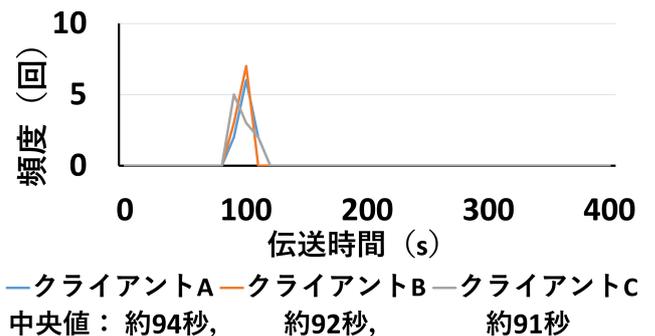


図 9 提案方式を適用させた時の伝送時間

## 7. 議論

本稿の提案では、サーバからクライアントにパケットロスしたシーケンス番号を送信している。しかし、どれくらい連続したパケットロスのシーケンス番号を送るかという事は考慮していない。これにより、ユニキャストで再送信中に、マルチキャストで再送信をし、パケットが重複す

る場合がある。この対策として、送信のマルチキャストと再送信のマルチキャストの間にタイムスロット期間を設け、その期間内にユニキャストを終了させ、全てのクライアントがマルチキャストのみの再送信に切り替える事で解決する [37]。このタイムスロット期間の算出方法を式 2 に示す [38]。R はタイムスロット期間である。T はサーバが send フェーズでマルチキャストをもちいて 1 つのパケットを送信し、次のパケットを送信するまでの時間である。d はサーバからクライアントに ACK を送信し、クライアントからサーバにその ACK を返答する最大の時間である。ACK はクライアントからサーバにパケットロスしたシーケンス番号を送信後、ACK を送信する。サーバは ACK が返答された時間を各クライアントごとに保持し、最大の時間を求める。

$$R = \frac{d}{T} \quad (2)$$

本稿の提案は、クライアントの数に応じたユニキャストを使用している。この時、クライアントの数が多くなれば、ユニキャストの使用数が増え、結果遅延の増加になる [39]。したがって、ユニキャストの使用数に上限がある。この上限の決め方は send フェーズでデータを送信前に決定する。まず、マルチキャストで ACK を全てのクライアントに送信する。次にクライアントはその ACK を返答する。サーバはその ACK の数を勘定する。その数を上限とする。上限を越えてしまった場合は、サーバがクライアントから受信した順に First In First Out (FIFO) を使用し解決する [40]。

本稿の提案では、クライアントのユニキャストの再送信の終了タイミングを考慮していない。これにより、パケットロスしたシーケンス番号を送る時間に影響がある。この対策として、サーバはクライアントに再送信する最後のパケットのシーケンス番号を事前に送信し、そのパケットを受け取ったら終了する。もし、そのパケットがロスした場合、クライアントは Retransmission Timeout (RTO) でタイムアウト時間を計算し、終了する [41]。

## 8. おわりに

課題は、マルチキャストの送信によって生じたパケットロスの再送信が送信時間の増加になることである。提案では、クライアントからサーバにパケットロスしたシーケンス番号をユニキャストで送信していない時間を利用し、サーバからクライアントにユニキャストで再送信する方法を定めた。評価実験にはファームウェアに見立てた 100KB のファイルを使用した。評価は、パケットを送信した後にマルチキャストのみで再送信した時と提案方式を使用した時の伝送時間を比較した。結果として、提案方式を使用した時の伝送時間はパケットを送信した後にマルチキャスト

のみで再送信した時と比べて平均で約 123 秒から約 92 秒の約 31 秒減少し、約 25.22% の減少率であった。

## 参考文献

- [1] Dudhe, P., Kadam, N., Hushangabade, R. M. and Deshmukh, M. S.: Internet of Things (IOT): An overview and its applications, *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, pp. 2650–2653 (online), DOI: 10.1109/ICECDS.2017.8389935 (2017).
- [2] Choi, S. and Lee, J.-H.: Blockchain-Based Distributed Firmware Update Architecture for IoT Devices, *IEEE Access*, Vol. 8, pp. 37518–37525 (online), DOI: 10.1109/ACCESS.2020.2975920 (2020).
- [3] Nadir, I., Mahmood, H. and Asadullah, G.: A taxonomy of IoT firmware security and principal firmware analysis techniques, *International Journal of Critical Infrastructure Protection*, Vol. 38, p. 100552 (online), DOI: <https://doi.org/10.1016/j.ijcip.2022.100552> (2022).
- [4] Sun, C., Xing, R., Wu, Y., Zhou, G., Zheng, F. and Hu, D.: Design of Over-the-Air Firmware Update and Management for IoT Device with Cloud-based RESTful Web Services, *2021 China Automation Congress (CAC)*, pp. 5081–5085 (online), DOI: 10.1109/CAC53003.2021.9727516 (2021).
- [5] Zandberg, K., Schleiser, K., Acosta, F., Tschofenig, H. and Baccelli, E.: Secure firmware updates for constrained iot devices using open standards: A reality check, *IEEE access*, Vol. 7, pp. 71907–71920 (2019).
- [6] El Jaouhari, S. and Bouvet, E.: Secure firmware Over-The-Air updates for IoT: Survey, challenges, and discussions, *Internet of Things*, Vol. 18, p. 100508 (2022).
- [7] El Heni, N. and Lagrange, X.: Multicast vs Multiple Unicast Scheduling in High-Speed Cellular Networks, *VTC Spring 2008 - IEEE Vehicular Technology Conference*, pp. 2456–2460 (online), DOI: 10.1109/VETECS.2008.542 (2008).
- [8] Nath, S., Wu, J. and Yang, J.: Optimum Energy Efficiency and Age-of-Information Tradeoff in Multicast Scheduling, *2018 IEEE International Conference on Communications (ICC)*, pp. 1–6 (online), DOI: 10.1109/ICC.2018.8422521 (2018).
- [9] Li, J., Zhou, Y. and Chen, H.: Age of Information for Multicast Transmission With Fixed and Random Deadlines in IoT Systems, *IEEE Internet of Things Journal*, Vol. 7, No. 9, pp. 8178–8191 (online), DOI: 10.1109/JIOT.2020.2981144 (2020).
- [10] Zheng, H. and Boyce, J.: An improved UDP protocol for video transmission over Internet-to-wireless networks, *IEEE Transactions on Multimedia*, Vol. 3, No. 3, pp. 356–365 (online), DOI: 10.1109/6046.944478 (2001).
- [11] Kumar, S. and Rai, S.: Survey on transport layer protocols: TCP & UDP, *International Journal of Computer Applications*, Vol. 46, No. 7, pp. 20–25 (2012).
- [12] Join, C., Fliess, M. and Chaxel, F.: Model-Free Control as a Service in the Industrial Internet of Things: Packet loss and latency issues via preliminary experiments, *2020 28th Mediterranean Conference on Control and Automation (MED)*, pp. 299–306 (online), DOI: 10.1109/MED48518.2020.9183277 (2020).
- [13] Jian, Z., Dinggang, W., Kun, H. and Jin, Y.: An Approach for Storage and Search of UDP Packet Data, *2012 International Conference on Computer Science and Electronics Engineering*, Vol. 2, pp. 603–607 (on-

- line), DOI: 10.1109/ICCSEE.2012.147 (2012).
- [14] Lee, T.-H., Xie, X.-S. and Chang, L.-H.: RSSI-based IPv6 routing metrics for RPL in low-power and lossy networks, *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 1714–1719 (online), DOI: 10.1109/SMC.2014.6974164 (2014).
- [15] Philopoulos, S. and Ferens, K.: Proxy-based connection-splitting architectures for improving TCP performance over satellite channels, *IEEE CCECE2002. Canadian Conference on Electrical and Computer Engineering. Conference Proceedings (Cat. No.02CH37373)*, Vol. 3, pp. 1430–1435 vol.3 (online), DOI: 10.1109/CCECE.2002.1012963 (2002).
- [16] Mtetwa, N. S., Tarwireyi, P., Abu-Mahfouz, A. M. and Adigun, M. O.: Secure Firmware Updates in the Internet of Things: A survey, *2019 International Multidisciplinary Information Technology and Engineering Conference (IMITEC)*, pp. 1–7 (online), DOI: 10.1109/IMITEC45504.2019.9015845 (2019).
- [17] Gnawali, O., Yarvis, M., Heidemann, J. and Govindan, R.: Interaction of retransmission, blacklisting, and routing metrics for reliability in sensor network routing, *2004 First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004.*, pp. 34–43 (online), DOI: 10.1109/SAHCN.2004.1381900 (2004).
- [18] Bohacek, S., Hespanha, J., Lee, J., Lim, C. and Obraczka, K.: A new TCP for persistent packet reordering, *IEEE/ACM Transactions on Networking*, Vol. 14, No. 2, pp. 369–382 (online), DOI: 10.1109/TNET.2006.873366 (2006).
- [19] Stojkoska, B. L. R. and Trivodaliev, K. V.: A review of Internet of Things for smart home: Challenges and solutions, *Journal of cleaner production*, Vol. 140, pp. 1454–1464 (2017).
- [20] Li, M., Gu, W., Chen, W., He, Y., Wu, Y. and Zhang, Y.: Smart home: architecture, technologies and systems, *Procedia computer science*, Vol. 131, pp. 393–400 (2018).
- [21] Almusaylim, Z. A. and Zaman, N.: A review on smart home present state and challenges: linked to context-awareness internet of things (IoT), *Wireless networks*, Vol. 25, pp. 3193–3204 (2019).
- [22] Hang, L. and Kim, D.-H.: Design and implementation of intelligent fire notification service using IP camera in smart home, *International Journal of Control and Automation*, Vol. 11, No. 1, pp. 131–142 (2018).
- [23] Yang, M.-J., Tham, J. Y., Wu, D. and Goh, K. H.: Cost effective IP camera for video surveillance, *2009 4th IEEE Conference on Industrial Electronics and Applications*, pp. 2432–2435 (online), DOI: 10.1109/ICIEA.2009.5138638 (2009).
- [24] Maier, A., Sharp, A. and Vagapov, Y.: Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things, *2017 Internet Technologies and Applications (ITA)*, IEEE, pp. 143–148 (2017).
- [25] Wang, F.: Neuro-WiFi: A novel neuronal connection underlies the potential interventional target, *Science Insights*, Vol. 43, No. 6, pp. 1179–1196 (2023).
- [26] Yoo, S., Shin, Y., Kim, S. and Choi, S.: Toward realistic WiFi simulation with smartphone “Physics”, *Proceeding of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks 2014*, IEEE, pp. 1–6 (2014).
- [27] Hai, L., Wang, H. and Wang, J.: Instantly decodable network coding for multiple unicast retransmissions in wireless point-to-multipoint networks, *IEEE Transactions on Vehicular Technology*, Vol. 65, No. 8, pp. 6232–6243 (2015).
- [28] Wang, W., Sun, S. and Wei, M.: Efficient Reliable Transmission for Mixed Multicast and Unicast Flows, *2013 International Conference on Computational and Information Sciences*, IEEE, pp. 1599–1602 (2013).
- [29] Lee, S. J., Tcha, Y., Seo, S.-Y. and Lee, S.-C.: Efficient use of multicast and unicast channels for multicast service transmission, *IEEE transactions on communications*, Vol. 59, No. 5, pp. 1264–1267 (2011).
- [30] Roy, V., Alam, M. I. and Alam, M. S.: Various use of cloud computing with the help of IP camera based on line, *International Journal of Computer Applications*, Vol. 157, No. 2, pp. 4–8 (2017).
- [31] Hintermaier, W. and Steinbach, E.: A system architecture for IP-camera based driver assistance applications, *2010 IEEE Intelligent Vehicles Symposium*, IEEE, pp. 540–547 (2010).
- [32] Yang, M.-J., Tham, J. Y., Wu, D. and Goh, K. H.: Cost effective IP camera for video surveillance, *2009 4th IEEE conference on industrial electronics and applications*, IEEE, pp. 2432–2435 (2009).
- [33] Puesche, A., Bothe, D., Niemeyer, M., Sachweh, S., Pohlmann, N. and Kunold, I.: Concept of Smart Building Cyber-physical Systems Including Tamper Resistant Endpoints, *2018 International IEEE Conference and Workshop in Óbuda on Electrical and Power Engineering (CANDO-EPE)*, IEEE, pp. 000127–000132 (2018).
- [34] Manske, A.: Conducting a vulnerability assessment of an IP camera (2019).
- [35] Jongboom, J. and Stokking, J.: Enabling firmware updates over LPWANS, *Embed. World Conf* (2018).
- [36] Lai, K. and Baker, M.: Measuring bandwidth, *IEEE INFOCOM’99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No. 99CH36320)*, Vol. 1, IEEE, pp. 235–245 (1999).
- [37] Bandi, A., Shankar, M. R. B., Chatzinotas, S. and Ottersten, B.: Joint multislot scheduling and precoding for unicast and multicast scenarios in multiuser MISO systems, *IEEE Transactions on Wireless Communications*, Vol. 21, No. 7, pp. 5004–5018 (2021).
- [38] Kleinrock, L. and Lam, S. S.: Packet-switching in a slotted satellite channel, *Proceedings of the June 4-8, 1973, national computer conference and exposition*, pp. 703–710 (1973).
- [39] Wang, L., Jin, Y., Kim, H. and Kim, E. J.: Recursive partitioning multicast: A bandwidth-efficient routing for networks-on-chip, *2009 3rd ACM/IEEE International Symposium on Networks-on-Chip*, IEEE, pp. 64–73 (2009).
- [40] Craig, T.: Building FIFO and priorityqueuing spin locks from atomic swap, Technical report, Technical Report TR 93-02-02, Department of Computer Science, University of ... (1993).
- [41] Kesselman, A. and Mansour, Y.: Optimizing TCP retransmission timeout, *Networking-ICN 2005: 4th International Conference on Networking, Reunion Island, France, April 17-21, 2005, Proceedings, Part II 4*, Springer, pp. 133–140 (2005).