International Journal of Communications, Network and System Sciences, 2026, 19(6), 91-96
https://www.scirp.org/journal/ijcns
ISSN Online: 1913-3723
ISSN Print: 1913-3715

Port Storming Self-Instantiating Protocol: Elective Disclosure

Anthony Brian Mallgren

University of the District of Columbia, Washington DC, USA
Email: anthony.mallgren@udc.edu

How to cite this paper: Mallgren, A.B. (2026) Port Storming Self-Instantiating Protocol: Elective Disclosure. *International Journal of Communications, Network and System Sciences*, **19**, 91-96.
https://doi.org/10.4236/ijcns.2026.196006

Received: May 1, 2026
Accepted: June 27, 2026
Published: June 30, 2026

Abstract

A method set for network disclosure and discovery may allow a more robust and simplified approach in building an information technology network. This specific proposal may outline componentization in such a way; one, a self-instantiating, a sufficiently self-encompassing utilization resource, rather than relying upon externalities, client may allow content substantiation, function implementation, and then tiered elective disclosure, an authorized exposure of data and/or information, in revealing content, through functionality. Two, which may be contained within the first component, a discovery mechanism, which may operate from a registry, or through a network port storming, a term which designates potential effect in indiscriminate port scanning, method, which may essential scan network port, and which may reveal an interface, which may allow such intrinsic functionality to consume external functionality and inductively coalesce content through such an interface. It may be noteworthy that while a hardware implementation paradigm has been proposed, that until if and when such may be released in a market, that a principle of functional tiering, a potential hierarchical approach of permissiveness, rather than only free form function, through me utilized in order to prevent system vulnerability, though some such principle may be known to those that may be familiar with development.

Keywords

Computer, Architecture, Network, Culture, Society

1. Introduction

As was covered by Farber, D., & Baran, P. (1977) [1], early in information technology networking, telephonic methodization was used in network connection. This resulted in a series of constraining principles. Due to the expensive nature of random dialing, a paradigm of client server connection seemingly became standard. This seemingly carried over into internet indexing and search, as was documented by Schneider, K. G. (1996) [2], where a central server is used to access search results. However, as more data coalesced, data obscurity became an issue. Common search terms could return millions or more results. It was said by Arcitecta Unveils Metadata-Based Data Protection (2023) [3] that the size of data would reach 100 zettabytes by 2025. One could imagine how this may scale in traditional search engines. As discussed by (Aaron L., *et al.*, 2024) [4], some new techniques in synthesis came to fruition, though presented issues of its own issues, such as hallucination, as described by van Haeren, L., Ravitharan, S., Murray, E., & Barrera, E. (2025) [5]. Rather than creating an intuitive solution from the base up, layered patch work was rather tangled in on itself. While the exact reasons for this are unknown to the observer, it would seem some people had some interest in having their intellectual property used in new solutions. For example, when a new development paradigm, e.g., that natural language should be used in interacting with a computer, rather than rebuilding the foundational aspects of operating systems, it seems that a shim was attempted in generating code based upon vast layered abstraction with many dependencies in trying to integrate this paradigm into development. A rather conversative approach in such a strategy was described by Ahmed, N., Shahbaz, M., Khalil, H., Javed, Z., Khalid, M., Khan, F. and Maroof, Z. (2025) [6], though if one may perform less academic research on "vibe coding", it may be more telling of actual dynamics.

The aim here is performing some discovery of new possibility.

2. Interfacing with Currently Existent

Due to the nature of adoption, it would seem that a viable solution with anything but the largest concerted backing of both public and private groups would be to interface with existing implementation. Some explanation of better-known implementation was performed by Simpson, W. R., & Foltz, K. E. (2016) [7]. Essentially, there are several known protocol. What may server an interest here is that a series of address and port (allowing multiple identifier for a single main address, which a computer may contain many) may be utilized in connection. Where one may elect registration, one may also scan, or even storm, one or more address and port, maybe over a series of time.

While some protocol are known, and maybe documented, these are usually not self-instantiating, *i.e.*, if not preinstalled on a computer, which, from an ethical perspective (many provider may attempt a liability disclaimer, or, if hosted, at least in America, by default (though a brief enquiry into European provider demonstrated they had a striking knowledge of what was supposed to be private data on their hardware), attempt to aggregate the data, e.g., into "AI" (data synthesis) models, et cetera, which may have copyright and infringement implication) may result in some undesired adversity, research must be performed, and, in the quickest path to functionality, installation of a pre-existing solution must be performed, or, a, documentation must reviewed, and a solution created.

What may be proposed here may be self-instantiating solution, *i.e.*, during a scan, or storm, a target may present at least an abstraction of function and data, which may be invoked. For example, if an address may be scanned, an initial response of, [interface]:[function1:[schema1],[schema2]],[function2], et cetera, of which, a function could be an installation of a solution. It may be noteworthy that a series of abstraction, or layering, could be performed. For example, if [machine1] may request privileged functionality, data, or information, from [machine2], [machine2], either by mean of automation, or manual intervention, may divulge additionality. Also, if [machine1] may serve a source to [machine2], [machine2] may add functionality, data, and/or information, and serve, with referential integrity, to [machine3], properly attributing credit. Of course, as mentioned in an article undergoing publication, market mechanization, such as monetary transaction, may be integrated into such process.

In developing a new system, rather than attempting relying upon what may be or become legacy protocol, it may be that an interface system may be better utilized in the process. For example, if one system utilizing a crawling discovered based upon a new protocol should need to traverse an existing network, it could be that a tunnel mechanism could be utilized between two network. Likewise, exposure via an existing protocol may be viable, either as an abstraction or a proxy. While it may be that performing a scan on the entire internet for a newly implemented protocol may not be a worthwhile solution (though, given some type of indicator which may designate a port may be self-instantiating, such as logging any response returned in querying a port) in an absolutely immediate future, given adoption may occur, such incentive may increase over time. Since it is relatively early in such a process, it may not be prudent to recommend any such definite solution, though if a port query may return a response, it may be worthy of examination.

In testing, ports such as HTTP and SMTP were scanned, and the result seemed to be that there was no response upon port connection, but rather expected some command, for which knowledge may depend upon externalities.

3. New Paradigm

As was said before, there were self-imposed constraints in layering upon previous work, that seemed serving social/financial incentive rather than pure functionality (again, layering upon existing technology, rather than implementing new paradigmatic features in more foundational layers, such as proposed here; for example, ethernet is a protocol that has been relied upon for decades, and still remains a foundational protocol; HTTP, SMTP, et cetera, all have been built upon what may be considered legacy technology, especially given the proposed comparison above; Ajit Karnik (2000) [8] covered some aspect of some tactical maneuvering performed by such an actor). While one may, and perhaps will, debate such advantages, here, a more purely technical proposal may be made.

Hardware may present a paradigm shift beyond current implement. What was also covered in an article currently undergoing publication, triangulation, as touched upon by Klimburg, A., Faesen, L., Verhagen, P., & Mirtl, P. (2020) [9], may be used in identifying the geolocation of a component, reasonably allowing ascertainment of, additionally, jurisdiction, et cetera. What may seem odd to purely technical folk, or those in a more ethical community, may be that this may have been locked away by private company, which seemingly may only respond under the threat of criminal prosecution, which may not be invoked by authorities, maybe unless they are held to a similar accountability nature.

Even in research of what may be considered cutting edge technology, such as Starlink, a satellite-involved communication system providing internet to including those without reasonable access to alternative solution, as described by Young, M., & Thadani, A. (2022) [10], may show that there are artificial constraint in trying enforcing of using existing intellectual property, unnecessarily so (e.g., obscuring receiver to receiver addressing, instead attempting enforcing abstraction to pre-existing protocol), and that more advanced functionality, *i.e.* triangulation, may not be made available to a subscriber.

For research & development, and initial solution, some conceptualization may be helpful. There may be an idea of inherent directionality. Rather than using three network installation, a component may utilize three receiver in attempting location identification. This may not be required, though may remain an option. There also may exist network installation, and allowing use of such technique to a subscriber. This may lay a foundation which further innovation may develop.

Protocol may be looked at as well. There may be opportunity for optimization at a hardware level, though perhaps that may be better left for other article, as hardware development may be actually explored, and further mechanization discovery may be performed. What may be worth noting here may be that similar self-imposed constraint may be identified, and such trade-off (e.g., social, economic, legal, et cetera) explored, deliberated upon, et cetera. Transmission technique, either existing, and/or new, either in sole, primary, ancillary, complementary, protagonistic, antagonistic, et cetera, modality may be explored, researched, developed, and produced. An initially assessment may reveal a somewhat liquid constraining dynamic if one may consider abandoning existing protocol in favor of innovation.

Beyond hardware implementation, which, again, may be better explored in a laboratory, software principle may, as a thesis, synthesis, and antithesis, interweave into hardware, be developed and new methodization found. A similar self-instantiating paradigm, to that of above, may be developed for component, and system discovery, *i.e.*, a device may present nothing only identification to allow documentation, buy may be invoked via a discovery protocol, which may allow, though not necessarily require, and should maybe should remain minimal, perhaps unless abstracted away into another component and/or system, though a focus should be placed on intuitive adoption, new standardization development. At the least, how do I cause a component and/or system to self-instantiated into a component and/or system. This may of course feed into what may have been previously presented as an additional paradigm in interfacing with pre-existing solution for adoption enhancement, including breadth, depth, and velocity, e.g., permeation in privilege, incentive, resource allotment, et cetera.

While the scope of this article may focus on introducing a self-instantiating discovery protocol, which, in an initialize phase, may simply be port query, documentation, and relevant functional interface manifest, schema, permissibility, both current, and any relevant potential authorization, and perhaps other parameter, with the caveat that it may be uninformed misdirection due to premature recommendation, it may be that upon digestion of such data, that a standard device map abstraction, either self-maintained, or otherwise, may be utilized to facilitate interaction. For example, if there is some graphical component, that mapping may occur between such discovery, query a device map repository, engage what functionality may be known, then notify some agent of any functional voids. For example, if there is a standard device interface, say a keyboard, that such functionality may be obscured into standard function, such as a keyPress, or for a mouse, click, function, which may then be used to invoke lower level device driver through a standard repository, as to begin mapping. Again, what may be relevant is that these be abstracted in a way that may be immediately usable, for example, even without facilitation, that someone may simply connect through a discovery protocol, and readily invoke execution without external resources. A taxonomy may be used to automate device interaction. For example, if there again, may be a keyboard, it could be that one may invoke such code from a client, *i.e.*, invoking a keyPress function, sending along a parameter. The standard taxonomical mapping may simply facilitate such a process. Again, this functional harness of sort may be self-maintained, where one may develop such an interface, or it may be hosted. Again, such a paradigm may be early in development, and this may undergo revision or refinement. Such abstraction may be performed beyond a component level, into a system, or network level, et cetera.

While privilege may be dynamic, it may be that a token, if not input (which may be defined by any party) into the system, may be generated and returned upon initial request, which may be reused in subsequent interaction, which may be federated for inter-systemization. There may be human intervention, or parameter based escalation. For example, a request may be made, and approved, such enhancement would be attributed to such a token, and used into authorization. Lifecycle management of such tokenization may be left to another article. Extrinsic failure may be logged through observable isolation. A recommendation was made in another article, accepted though pending publication, for which component isolation was advocated for in ensuring that execution did not affect core system functionality, and, as a means for monitoring functionality, could be used to monitor and provide auditable information, both systemic and network, for any execution. Due to the nature of current networking, unless industry or government may bear the burden of network management, allowing gating, it may be that an audit and prosecution/prosecuting model may have to be relied upon.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

[1] Farber, D. and Baran, P. (1977) The Convergence of Computing and Telecommunications Systems. *Science*, **195**, 1166-1170. https://doi.org/10.1126/science.195.4283.1166

[2] Schneider, K.G. (1996) Internet Librarian. *American Libraries*, **27**, 66-67. http://www.jstor.org.udc.idm.oclc.org/stable/25633950

[3] (2023) Arcitecta Unveils Metadata-Based Data Protection. *Computer Security Update*, **24**, 1-4. https://www.jstor.org/stable/48711130

[4] Aaron, L., Abbate, S., Allain, N.M., Almas, B., Fallon, B., Gavin, D., *et al.* (2024) AI Literacy. In: Aaron, L. and Gavin, D., Eds., *Optimizing AI in Higher Education: SUNY FACT² Guide, Second Edition*, State University of New York Press, 18-23. https://doi.org/10.2307/jj.20522984.15

[5] van Haeren, L., Ravitharan, S., Murray, E. and Barrera, E. (2025) Challenges. In: Campagnolo, Y., Ed., *Artificial Intelligence's Impact on Legal Journals/Incidence de l'intelligence artificielle sur les revues de droit: Challenges and Opportunities for the Ottawa Law Review/Défis et possibilités pour la Revue de droit d'Ottawa*, University of Ottawa Press, 13-20. https://doi.org/10.2307/jj.26248835.5

[6] Ahmed, N., Shahbaz, M., Khalil, H.S.U.R., Javed, Z., Khalid, M.A., Khan, F.A., *et al.* (2025) The Synergy of Low-Code/No-Code and AI/ML: Enhancing Intelligent Automation. *World Journal of Engineering and Technology*, **13**, 754-763. https://doi.org/10.4236/wjet.2025.134048

[7] Simpson, W.R. and Foltz, K.E. (2016) Enterprise Considerations for Ports and Protocols. Institute for Defense Analyses. http://www.jstor.org/stable/resrep22700

[8] Karnik, A. (2000) Microsoft, Competition and Competition Policy. *Economic and Political Weekly*, **35**, 2949-2958. http://www.jstor.org/stable/4409623

[9] Klimburg, A., Faesen, L., Verhagen, P. and Mirtl, P. (2020) Pandemic Mitigation in the Digital Age. Austrian Institute for European and Security Policy. https://www.aies.at

[10] Young, M. and Thadani, A. (2022) Managing Large LEO Constellations. In: Young, M. and Thadani, A., Eds., *Low Orbit, High Stakes: All-In on the LEO Broadband Competition*, Center for Strategic and International Studies (CSIS), 18-23. http://www.jstor.org/stable/resrep47096.8