Problem Analysis

Large organisations require fast and efficient network monitoring systems, and many current systems use slow or difficult to set up methods for informing persons of issues such as services offline or environment issues[1]. Security is also a relevant factor, but dealing with potential security breaches or even methods of working out if they are happening are conspicuously missing from many reports and documents[2]. A paper on visualising Cyber Security is very much based around post-analysis, though its favour of colourful visualisations is not shared by many Cyber Security analysts in the field, who would much rather work with “archaic” command line tools[3]. This lack of interest in the visualisation of data makes it more difficult to identify trends in live data, as traversing textual logs can be both time consuming and frustrating[4].

Due to the unification of operating systems over devices, such as the Universal Windows Platform or many Linux variants and Android sharing similar codebases, cross-platform attacks are more popular than ever[5]. This means that monitoring and intrusion detection systems should be take this into account, keeping an eye on as much of the site as possible and not just servers or desktops.

The idea behind Natural Language Processing is to convert text into a programmer friendly data structure that describes the meaning of the text, allowing it to be used programmatically[6].

Chat bots started out as fairly non-intelligent programs, able to only parse very basic and very specific text out of incoming data[7]. Even the more intelligence chat bots such as Eliza and ALICE are still just simple adaptations of the “regular expression” style of chat bots, often written in a style of XML called AIML[8]. While you could still have almost meaningful or semi-useful information gathering conversations with chatbots in the early days of them[8][9], things slowly started to improve with projects taking advantage of large databases of documented human interaction such as the Dialog Diversity Corpus[10]. These databases allowed for much larger keyword recognition and transformations of incoming text to output human-like responses[11]. However, while conversations seemed more natural, this was still only a basic extension of the original “regular expression” style.

As machine learning started to enter the realm of computer science problems, it started to merge with the idea of being able to process natural language, and that began the field of statistical natural language processing[12]. Machine learning allowed chat bots to take interactions like the Dialog Diversity Corpus and learn from them, being able to “guess” at new phrases or sentences rather than just having a list of stock inputs and outputs. Machine learning is also not bound by its reference material, and is able to learn from the inputs it receives(ref?).

Attempts have been made to start the idea of natural language processing and machine learning almost from scratch, but so far it seems that it almost always worth referring back to technologies that already exist and building upon them[6]. For example, a study comparing traditional AIML with the ALICE chat bot and a new chat bot program called FUTURE showed that the AIML based bot was significantly easier and faster to develop for due to already having many AI-oriented features, and ended up performing fairly well under human testing[13].

“Machine Intelligence” was defined in one of the papers as a list of fundamental requirements for a machine to be considered “intelligent”, such as having perception of a situation, working sensors and a working memory that can be used to retrieve data logically. Many chat bots can be considered partially intelligent due to being able to apply logic & reasoning, heuristics and a dynamic memory[13].

Recently, technologies like Microsoft’s Language Understanding Intelligent Service (LUIS)[14] allow for non-machine learning experts build models for things like natural language processing without complex programming or large-scale database management[15]. Machine learning can suffer from what as being a “lopsided” problem, where not all the data is available for a system to fully realise a piece of data. A human “teacher” can help rectify this and act as the middle-man, translating language easily into machine-readable code until the machine has learned enough to continue for itself[15]. After systems like these achieve the desired accuracy, they can be accessed via external API’s which will accept an input and output a much more machine-useable output, with information separated as requested by the teacher[16].

This table briefly compares IT Infrastructure monitoring tools to see where they get their information, how they notify administrators of potential issues, and if they can be scaled past a small set of servers.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name of Tool** | **OS Support** | **Data Sources** | **Notification Methods** | **Scalability** |
| Nagios[17] | Windows, any REHL Linux and most other OS’s[18] | Servers send log data via JSON to log server[19]  Automated host/service checks[20] | Email  SMS  Pager  Windows Popup  Yahoo/ICQ/MSN  Audio alert  Web-based  Custom third-party packages[21] | Small-medium size infrastructure, not suited for large or extremely large corporations.[22] |
| Webmin[23] | Linux Only[23] | Installed on every server, gathers information. Can be clustered to manage multiple servers at once.[24] | Email  Can get notifications via API[25]  Web-based | Small scale only[26] |
| Microsoft Remote Server Administration Tools[27] | Windows Only | Plug-in tools to control remote servers, uses Windows Management Instrumentation and Powershell to gather data automatically | Tool-based only | Small scale only |
| Icinga[28] | Windows, any REHL Linux and most other OS’s[29] | Servers send log data to log server  “Satellite” monitoring servers provide data to the master logging server[29]  Automated host/service uptime checks | Provides addons for notifications to almost any service, and allows templating to add new services[30]  Provides a query-able API | Large scale support, with distributed log servers and high customisability in terms of notifications[31] |
| OpenNMS[32] | REHL Linux, Debian, Windows, Vagrant | Runs regular tests against services to check uptime | XMPP  SNMP  Email  Paging  Arbitrary HTTP GET/POST’s[33] | Designed to be split across multiple nodes – Very scalable if managed properly[34] |

This table shows the advantages and disadvantages of various notification methods from the previous list of tools. This is assuming a tool has picked up on the issue and is trying to contact a person able to attend to it.

|  |  |  |
| --- | --- | --- |
| **Notification Method** | **Advantages** | **Disadvantages** |
| E-Mail | Can provide a lot of information at once  Can format information well  Is almost instantaneous  Uses service probably already paid for by the company  Can receive commands in response | Unlikely to be seen immediately due to lack of push notifications on most email services |
| SMS | Fast and cheap  Likely to be seen quickly, even if the receiver has no internet  Can receive commands in response | Limited information able to be given  Would likely have to use outside services |
| Phone Call (Automated) | Gets attention of receiver immediately if phone is on | Limited information able to be given  Receiver has to actively answer their phone to receive information |
| Web page | Can provide a lot of information at once  Can format information well  Is almost instantaneous  Can receive commands in response | Must be checked manually |
| Log files | Contains all information | Information could be badly formatted  Must be checked manually |
| Instant Message(XMPP or other) | Can provide a medium amount of information  Can format information in some cases, depending on service  Is almost instantaneous – Push notifications available through major platforms  Likely to be seen quickly  Can receive commands in response | Requires receiver to have internet  Requires receiver to have IM service application installed |

This table shows advantages and disadvantages of various response types – How a receiver of a notification can then affect a problem with their infrastructure while off site.

|  |  |  |
| --- | --- | --- |
| **Response** | **Advantages** | **Disadvantages** |
| Remote desktop | Can see whole server and access tools or logs | Requires VPN or public facing remote desktop server  Laggy/unstable connection can cause issues in operating server |
| SSH | Can see whole server and access tools or logs | Requires VPN or public facing SSH server  Can’t see graphical information, might be hard to understand data at a glance |
| Responding to notification (Email, IM, SMS) | Instant action based on the request | Requires receiver of response to be able to understand and act upon request |
| Web page | Can see and act upon information on the page  Instantaneous | Requires VPN or public facing web server |

At this point, there are many working IT infrastructure monitoring tools, and this is not where the issue lies. The issue lies in informing the right people of the right information, and giving them the tools to deal with the information quickly.

While many of the existing tools show promise in their ability to send notifications to remote users, via a variety of different methods, many of them then require a large number of steps to attempt to rectify an issue. If an administrator gets a notification via SMS, they must then get onto a PC, VPN into their workplace, SSH into the problem server and can only then begin to figure out the scope of the problem. This is assuming the notification made it to them in the first place, they were able to read it in a timely manner, they are able to get to a work-station, and if notifications were even set up correctly.

Many of the tools rely on technology that is either not suited to the task or out of date. For example, sending an email with an issue report – It could be hours before the receivers phone checks in to the mailserver, and even then requires the receiver to drop whatever they are doing to respond to the issue.

The major issues highlighted in this analysis are firstly the difficulty of connecting infrastructure monitoring services to notification services, and secondly the inability for infrastructure maintainers to reliably manage and fix issues in their infrastructure remotely in a timely manner.

# References

[1] and M. I. B. Rafiullah Khan, Sarmad Ullah Khan, Rifaqat Zaheer, “An Efficient Network Monitoring and Management System,” *Int. J. Inf. Electron. Eng.*, vol. 3, no. 1, p. 122, 2013.

[2] N. Chandrika, “Cyber Security in the UK,” *POSTnote*, no. 389, pp. 1–4, 2011.

[3] G. A. Fink, C. L. North, A. Endert, and S. Rose, “Visualizing cyber security: Usable workspaces,” in *6th International Workshop on Visualization for Cyber Security 2009, VizSec 2009 - Proceedings*, 2009, pp. 45–56.

[4] H. Shiravi, A. Shiravi, and A. a. Ghorbani, “006 A survey of visualization systems for network security,” *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 8, pp. 1313–1329, 2012.

[5] R. Sharma, “Study of Latest Emerging Trends on Cyber Security and its challenges to Society,” *Int. J. Sci. Eng. Res.*, vol. 3, no. 6, 2012.

[6] R. Collobert, J. Weston, and L. Bottou, “Natural language processing (almost) from scratch,” *J. Mach. …*, vol. 12, pp. 2493–2537, 2011.

[7] B. A. Shawar and E. Atwell, “Chatbots: are they really useful?,” *LDV-Forum*, 2007.

[8] R. S. Wallace, “The Anatomy of A.L.I.C.E.” [Online]. Available: http://www.alicebot.org/anatomy.html.

[9] J. Rahman, “Implementation of ALICE chatbot as domain specific knowledge bot for BRAC U (FAQ bot),” 2012.

[10] “Dialogue Diversity Corpus.” [Online]. Available: http://www-bcf.usc.edu/~billmann/diversity/DDivers-site.htm. [Accessed: 15-Dec-2016].

[11] B. A. Shawar and E. Atwell, “Using dialogue corpora to train a chatbot,” 2003.

[12] P. M. Nadkarni, L. Ohno-Machado, W. W. Chapman, C. Manning, P. Raghavan, H. Schuetze, W. Hutchins, N. Chomsky, A. Aho, R. Sethi, J. Ullman, N. Chomsky, J. Friedl, S. Kleene, B. Kernighan, R. Pike, J. Levine, T. Mason, D. Brown, A. Joshi, K. Vijay-Shanker, D. Weir, W. Clocksin, C. Mellish, D. Warren, D. Klein, N. Chomsky, D. Klein, C. Manning, J. Quinlan, D. Klein, C. Manning, U. of Pennsylvania, C. Manning, H. Schuetze, D. Jurafsky, J. Martin, P. Spyns, L. Deleger, F. Namer, P. Zweigenbaum, J. Denny, A. Spickard, K. Johnson, S. Haas, G. Savova, J. Masanz, P. Ogren, A. Aronson, Q. Zou, W. Chu, C. Morioka, H. Liu, A. Aronson, C. Friedman, T. Rindflesch, A. Aronson, T. Pedersen, M. Weeber, J. Mork, A. Aronson, W. Chapman, W. Bridewell, P. Hanbury, P. Mutalik, A. Deshpande, P. Nadkarni, Y. Huang, H. Lowe, W. Chapman, W. Bridewell, P. Hanbury, U. of S. (Hungary), G. Savova, W. Chapman, J. Zheng, C. Tao, H. Solbrig, S. Deepak, G. Hripcsak, N. Elhadad, Y. Chen, R. Taira, D. Johnson, V. Bhushan, N. Sager, M. Lyman, N. Nhan, P. Haug, D. Ranum, P. Frederick, L. Christensen, P. Haug, M. Fiszman, H. Xu, C. Friedman, P. Stetson, L. Christensen, H. Harkema, J. Irwin, O. Uzuner, B. South, S. Shen, O. Uzuner, I. Goldstein, Y. Luo, O. Uzuner, O. Uzuner, I. Solti, E. Cadag, C. Chute, L. Chen, C. Friedman, X. Wang, G. Hripcsak, C. Friedman, W. Chapman, M. Fiszman, J. Dowling, W. Chapman, J. Dowling, M. Wagner, X. Wang, G. Hripcsak, M. Markatou, D. Lindberg, B. Humphreys, A. McCray, A. Browne, A. McCray, S. Srinivasan, G. Divita, A. Browne, T. Rindflesch, N. L. of Medicine, S. Srihari, C. Elkan, M. Hearst, S. Dumais, E. Osman, D. Inc., D. Fradkin, I. Muchnik, A. Viterbi, A. Dempster, N. Laird, D. Rubin, M. Hasegawa-Johnson, J. Zhang, D. Shen, G. Zhou, E. Sonnhammer, S. Eddy, E. Birney, A. Lukashin, M. Borodovsky, S. Eddy, L. Rabiner, J. Lafferty, A. McCallum, F. Pereira, A. Culotta, D. Kulp, A. McCallum, C. Sutton, A. McCallum, A. Franz, T. Brants, U. of S. N. L. Group, A. Foundation, A. Foundation, C. Thompson, M. Califf, R. Mooney, N. A. C. of the A. for C. Linguistics, P. Fodor, A. Lally, D. Ferrucci, A. Lally, P. Fodor, L. Libresco, K. Jennings, J. Fitzgerald, E. Berner, G. Webster, A. Shugerman, C. Friedman, A. Elstein, F. Wolf, R. Miller, F. Masarie, and R. Miller, “Natural language processing: an introduction.,” *J. Am. Med. Inform. Assoc.*, vol. 18, no. 5, pp. 544–51, 2011.

[13] A. Khanna, B. Pandey, K. Vashishta, K. Kalia, B. Pradeepkumar, and T. Das, “A Study of Today’s A.I. through Chatbots and Rediscovery of Machine Intelligence,” *Int. J. Sci. Technol.*, vol. 8, no. 7, pp. 277–284, 2015.

[14] Microsoft, “LUIS: Language Understanding Intelligent Service (beta).” [Online]. Available: https://www.luis.ai/. [Accessed: 15-Dec-2016].

[15] P. Simard, D. Chickering, A. Lakshmiratan, D. Charles, L. Bottou, C. Garcia, J. Suarez, D. Grangier, S. Amershi, J. Verwey, and J. Suh, “ICE: Enabling Non-Experts to Build Models Interactively for Large-Scale Lopsided Problems,” 2014.

[16] J. D. Williams, E. Kamal, M. Ashour, H. Amr, J. Miller, and G. Zweig, “Fast and easy language understanding for dialog systems with Microsoft Language Understanding Intelligent Service (LUIS),” 2015.

[17] Nagios, “Nagios - The Industry Standard In IT Infrastructure Monitoring,” 2017. [Online]. Available: https://www.nagios.org/. [Accessed: 25-Feb-2017].

[18] Nagios, “Operating Systems - Nagios Exchange,” 2017. [Online]. Available: https://exchange.nagios.org/directory/Plugins/Operating-Systems. [Accessed: 25-Feb-2017].

[19] Nagios, “Nagios Log Server - Monitoring a New Log Source,” 2016.

[20] Nagios, “Service Checks · Nagios Core Documentation,” 2017. [Online]. Available: https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/servicechecks.html. [Accessed: 25-Feb-2017].

[21] Nagios, “Notifications · Nagios Core Documentation,” 2017. [Online]. Available: https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/notifications.html. [Accessed: 25-Feb-2017].

[22] Opsview, “Why Scalability is Such a Big Concern When Using Nagios | Opsview,” 2016. [Online]. Available: https://www.opsview.com/resources/blog/why-scalability-such-big-concern-when-using-nagios. [Accessed: 25-Feb-2017].

[23] Webmin, “Webmin,” 2017. [Online]. Available: http://webmin.com/. [Accessed: 25-Feb-2017].

[24] Webmin, “Cluster Webmin Servers - Webmin Documentation,” 2017. [Online]. Available: http://doxfer.webmin.com/Webmin/Cluster\_Webmin\_Servers. [Accessed: 25-Feb-2017].

[25] Webmin, “API Webmin - Webmin Documentation,” 2017. [Online]. Available: http://doxfer.webmin.com/Webmin/API\_Webmin#show\_webmin\_notifications.28.5Bno-updates.5D.29. [Accessed: 25-Feb-2017].

[26] Jack Wallen, “Webmin: One big drawback to using this data center management tool - TechRepublic,” 2016. [Online]. Available: http://www.techrepublic.com/article/webmin-one-big-drawback-to-using-this-data-center-management-tool/. [Accessed: 25-Feb-2017].

[27] Microsoft, “Remote Server Administration Tools (RSAT) for Windows operating systems,” 2016. [Online]. Available: https://support.microsoft.com/en-gb/help/2693643/remote-server-administration-tools-rsat-for-windows-operating-systems. [Accessed: 25-Feb-2017].

[28] Icinga, “Icinga – Open Source Monitoring,” 2017. [Online]. Available: https://www.icinga.com/. [Accessed: 25-Feb-2017].

[29] Icinga, “Service Monitoring :: Icinga Web,” 2017. [Online]. Available: https://docs.icinga.com/icinga2/latest/doc/module/icinga2/toc#!/icinga2/latest/doc/module/icinga2/chapter/service-monitoring#service-monitoring-windows. [Accessed: 25-Feb-2017].

[30] Icinga, “icinga2 Documentation - Notification Script and Interfaces,” 2017. [Online]. Available: https://docs.icinga.com/icinga2/latest/doc/module/icinga2/toc#!/icinga2/latest/doc/module/icinga2/chapter/addons#notification-scripts-interfaces. [Accessed: 25-Feb-2017].

[31] C. Haen, E. Bonaccorsi, and N. Neufeld, “DISTRIBUTED MONITORING SYSTEM BASED ON ICINGA,” 2011.

[32] OpenNMS, “Installation |,” 2017. [Online]. Available: https://www.opennms.org/en/install. [Accessed: 25-Feb-2017].

[33] OpenNMS, “Configuring notifications - OpenNMS,” 2017. [Online]. Available: https://wiki.opennms.org/wiki/Configuring\_notifications. [Accessed: 25-Feb-2017].

[34] Jeff Gehlbach, “Scalability of opennms,” 2008. [Online]. Available: http://opennms-discuss.narkive.com/wIRXrWZk/scalability-of-opennms. [Accessed: 25-Feb-2017].