Assessment of deterministic delay bounds for a DoS-attack prevention device with a static window flow control

Vitaly G. Promyslov*, Kirill V. Semenkov**

** Institute of Control Sciences of Russian Academy of Sciences,
Moscow, Russia (e-mail: v1925@mail.ru).
** Institute of Control Sciences of Russian Academy of Sciences,
Moscow, Russia (e-mail: semenkovk@mail.ru)*

Abstract: This article focuses on the justification of the timing characteristics of devices using a static window flow control mechanism for protection against denial of service (DoS) attacks. The main focus is on a particular type of DoS attack, like flood and hit-and-run attacks. An attack of that kind can be performed using the black-box approach with a minimal piece of knowledge about the internals of the attacked system. The methods of tropical algebra (min-plus algebra) are used to compute the timing characteristics of the device with static window flow control.

Keywords: control and security for critical systems, network calculus, denial of service, systems with time-delays

1. INTRODUCTION

The digital revolution that currently is underway in the society and industry did not leave aside relatively conservative areas like industrial control systems**.** During the last decades, they went through the transformation from mainly analog to digital systems and further, according to the Industry 4.0 concept, to open cyberphysical systems interacting via the Internet. The advantages of the transformation are evident, but the late changes have their negative sides. In particular, this dark side is the increasing risk of cyberattacks on control systems. The impact on the control system may impede the functioning of the attacked system and damage, put out of order the control object itself.

The risks are very high for the control systems of critical backbone objects like power supply systems, transportation infrastructure and autonomous vehicles, oil and gas industrial object, nuclear power plants. A kind of possible cyber-impact to a control system is the so-called denial-of-service (DoS) attack. The data from the recent review (DDoS 2019) shows the high rise of the DoS-attacks incidents: up to 17 percent for the last year.

From the attacker's point of view, the DoS attacks have a set of advantages before other cyber attacks: an intruder may get only external access to the attacked system; they are highly scalable and can be performed remotely. The main goal of the DoS attack is to make the system inaccessible to users, block its operation. On the other hand, an attacker may use a DoS-attack to divert attention from other harmful effects. When the actions of the attackers reach their goal, a system owner or a user can instantly detect it by the malfunctioning of the system or fault of a resource located there. However, some indirect signs allow detecting a DoS attack from the very beginning.

The first sign of a DoS-attack is a sharply increased system load that reflects in rising the system accessibility time that considerably differs from the average normal operation response time or the rapid increase in incoming traffic on one or several ports.

The other signs might be that the system begins to frequently crash, freeze, shut down incorrectly, et cetera.

In the paper, we are going to consider the possibility of modeling the impacts on that kind of attack on an industrial control system. We will describe algorithms that a protected system may use to counter the attacks. The mathematical model description uses the "Network Calculus" method, and section 2 is a short introduction to the mathematical basement of the method.

2. THE NETWORK CALCULUS

The Network Calculus (see Le Boudec and Thiran, 2019) is a relatively new area of applied mathematics invented by Cruz (1991a, 1991b) and based on min-plus algebra (see Baccelli, 1992). The application area of the Network Calculus is the research of queuing systems. The general area of the Network Calculus application is calculations of computational network and system characteristics (see examples of the usage in Masolkin and Promyslov (2010) and Baybulatov and Promyslov (2017), He and Li (2017). The key aspect of the method is the usage of deterministic restriction of the flow that allows the work with a broad set of incoming flow types and

The reported study (Section III) was partially funded by RFBR, project number 19-29-06044

makes the theory easy-to-use and applicable to practical engineering problems (see Baybulatov and Promyslov, 2019).

Below we will briefly introduce the Network Calculus. For the advanced reading, we would recommend the book by Le Boudec and Thiran (2019).

Definition 1: Flow function (cumulative flow function) is a wide-sense increasing function of time:

$$\begin{cases} A(t) \leq A(s), \forall t < s \\ A(t) \in \mathbb{R}_+ \cup \{+\infty\} \\ t \in \mathbb{R} \end{cases}$$

Let us name a flow function causal if $A(t) = 0, \forall t \leq 0$. Now we define convolution and deconvolution operations.

Definition 2: Consider two causal flow functions A and β. Their convolution designated as $\otimes$ is the function $A^* = A \otimes \beta$ is:

$$A^*(t) = \inf_{0 \leq s \leq t} \{\beta(t-s) + A(s)\}$$

Further, we will omit argument t in the equations, if it is not necessary. It is obvious that $A^*(t) = 0, \forall t < 0$, and A^* is non-negative because both A and β are non-negative and causal.

Definition 3: Consider two flow functions A and β, where β is causal; deconvolution operation designated as $\oslash$ is the function $H = A \oslash \beta$:

$$H = \sup_{u \geq 0} \{A(t+u) - \beta(u)\}$$

Note that deconvolution H of flow functions A and β, where β is causal is a flow function itself.

Definition 4: A function β is the (minimal) service function of a network element (or system) with the input flow A if β is a causal flow function and the element (system) output flow A^* satisfies $A^* \geq A \otimes \beta$.

To describe data flows within a system, the Network Calculus generally does not use flow function directly but instead uses a "derived" function called flow envelope (or flow arrival curve). The flow envelope evaluates the flow scale, and it was invented by Cruz R.L. (1991a).

Definition 5: A function a is the envelope of flow A if $A \leq A \otimes a$ or, that is the same, $a \geq A \oslash A$.

For linear systems with input flow A, output flow A^*, $A(t) \geq A^*(t)$, the following equation ties input flow envelope a, system or system component service function β, and output flow envelope a_*:

$$a_* = a \oslash \beta$$

See the proof in Le Boudec and Thiran (2019).

Now we introduce a useful function $\delta_T(t)$ that is named burst-delay function:

$$\delta_T(t) = \begin{cases} +\infty, t \geq T \\ 0, t < T \end{cases}, \quad \text{and } T \geq 0$$

Definition 5 *(Sub-additive closure)*: Let a function $f: \mathbb{R} \to \mathbb{R}_+ \cup \{+\infty\}$. Denote as $f^{(n)}$ the function obtained by $(n-1)$ times repeating convolutions of f with itself and state that $f^{(0)} = \delta_0$ (the burst-delay function). Then the sub-additive closure $\bar{f}$ of f is

$$\bar{f} = \inf_{n \geq 0} \{f^{(n)}\}$$

or

$$\bar{f} = \delta_0 \wedge f \wedge (f \otimes f) \wedge \ (f \otimes f \otimes f) \wedge \ldots$$

Definition 6 *(The maximal delay in the system)*: For linear systems with input flow A, the output flow A^*, $A(t) \geq A^*(t)$, the maximal delay D_{max} expressed as maximal horizontal distance between input and output flow curves is:

$$D_{max} = h(A, A^*) = \sup_{t \geq 0} \{\inf\{d \geq 0: A(t) \leq A^*(t+d)\}\} \quad (1)$$

and it is easy to show that

$$D_{max} = \tilde{h}(A, A^*) = \inf\{d \geq 0: A \oslash A^*(-d) \leq 0\} \quad (2)$$

3. MATHEMATICAL MODEL OF A SYSTEM UNDER ATTACK

3.1 Introduction to DoS-attack formalism

An attack of denial-of-service type is an attack to a computing system with the purpose of to transfer the system to the "out of service" state, that is, an attack that tries creating the conditions that disable or impede the access to the system resources (usually, servers) for legitimate users. Currently, they might be divided into four types:

1. The flood attacks focused on data channel overflow, or brute force attack. Their goal is to jam the channel of the system under attack and break the system availability for legitimate users.
2. Attacks using vulnerabilities of the network protocol stack.
3. Application-level attacks.
4. Attacks of the indirect kind, that is, attacks going to activate false-positive protection measures in so make the resource unavailable.

This paper deals with the first type of attack only. This type of attack can target different functional properties and protocols of the system like http, ping, flood, smurf, and fraggle attack, et cetera. The DoS attacks can be categorized according to the number of offsprings launching the attack and the techniques used to conduct it. Unlike other attacks, they have a general character because an attacker does not need a deep understanding of an attacked system. In the paper, we do not consider the exact type of DoS-attack but for simplicity refer to single source flooding.

3.2 Mathematical description of a DoS-attack and the reaction of the attacked system, algorithms of attack countering

The general scheme of the DoS-attack model regulator is shown in Fig.1. This example closely relates to the results obtained by Chang. (1998) and Agrawal et al. (1999).

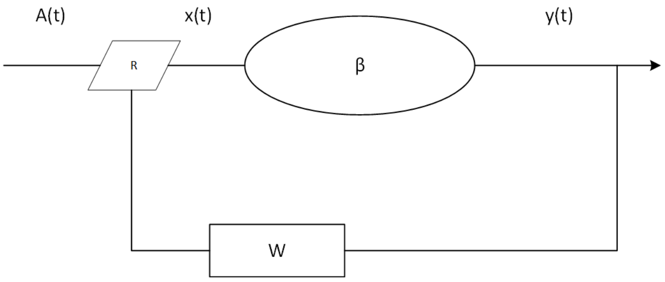

Fig. 1. DoS-attack prevention and detection scheme.

Here R is a clipper serving as a DoS-attack prevention device that effectively restricts the flow if backlog in the system exceeds the limit W. This device might be viewed as a "light" greedy shaper that indirectly controls properties of the input flow. It is a device delaying input bits in the buffer that is large enough, whenever sending a bit would violate the constraint W but outputs them as soon as possible. Le Boudec and Thiran (2019) made an in-depth analysis of the model.

Let a data flow $A(t)$ constrained by the envelope $a(t)$ passes through a flow clipper device to the system offering service curve β. The window flow clipper limits the amount of data passed to the network in such a way that the total backlog is less than or equal to a constant $W > 0$, where W is a static window.

We set a problem of impact estimation of the used flow clipper device with a window W on the timing characteristics of the system via the delay value. Using (1) directly define the maximal delay value D_{max} as

$$D_{max} = h\{x_{max}, y_{min}\} \qquad (3),$$

where x_{max} is maximal flow after flow clipper and y_{min} is minimal output flow.

3.3 System definition

Now let us consider a model of the system shown in Fig. 1 with the input flow $A(t) \geq 0\,, t \geq 0$. The system after the prevention device performs the transformation of the flow x to the flow y:

$$\Pi : x \rightarrow y = \Pi(x) \qquad (4)$$

or

$$\Pi(x) \geq \mathsf{C}_\beta(x) \qquad (5)$$

Where C_β is a convolution operator (Le Boudec, J-Y., and Thiran, P. (2019). We do not consider an exact mapping, but we know that

$$y(t) \geq (\beta \otimes x)(t) \qquad (6)$$

$$\begin{cases} x(t) \leq A(t) \\ x(t) \leq \Pi(x) + W \end{cases} \qquad (7)$$

$$y(t) \geq A(t - D) \qquad (8)$$

where D is some big enough delay. Solving the inequalities (7) and applying the theorem 4.3.1 from Le Boudec and Thiran (2019) one gets:

$$x_{max} = \overline{(\Pi + W)}(A) \geq \overline{\left(\mathsf{C}_\beta + W\right)}(A) = \overline{\left(\mathsf{C}_{\beta+w}\right)}(A) = \mathsf{C}_{\overline{\beta+w}}(a) = \overline{(\beta + W)} \otimes A \qquad (9)$$

It follows from the equations (5, 6) and (9) that

$$y(t) \geq (\beta \otimes x)(t) = (A \otimes \beta_{cl})(t) \qquad (10)$$

and then

$$\beta_{cl} = \beta \otimes \tilde{\beta}_w \qquad (11)$$

where $\tilde{\beta}_w = \overline{(\beta + W)}$ and β_{cl} is a system closed-loop service function.

3.4 Working delay

The delay when the system's backlog is near but not empty sets the maximal working value of the delay that is an important reference point for system developer. Using the equation (3) and taking into account that $\overline{y_{mın}(t)} = A(t - D)$, see (8), and $\overline{x_{\max}(t) =}\ \overline{(\beta + W)} \otimes A$ is defined by (9) we can get bound for delay:

$$D_{work} = h(\overline{y_{mın}}, \overline{x_{\max}}) = h(A, \tilde{\beta}_w \otimes A\ ) = \sup_{t \geq 0} \inf\{D \geq 0\ such\ A(t) \leq \left(\tilde{\beta}_w \otimes A\right)(t + D)\} \qquad (12)$$

Let us rewrite (12) as

$$D_{work} = \inf\left\{D \geq 0\ such\ A \oslash \left(\tilde{\beta}_w \otimes A\right)(-D)\ \leq\ 0\right\} = \inf\{D \geq 0\ such\ (A \oslash A) \oslash \left(\tilde{\beta}_w\right)(-D)\ \leq\ 0\} \qquad (13)$$

Combining (2) and (13) we get

$$D_{work} = \tilde{h}\left(A \oslash A, \tilde{\beta}_w\right)$$

Now let us remember that minimal flow envelope $a = A \oslash A$, so

$$D_{work} = \tilde{h}\left(a, \tilde{\beta}_w\right) \qquad (14)$$

Consider an important case of (12) when $(\beta + W)$ is a sub-additive function (i.e. $f(t + s) \leq f(t) + f(s)$ for any $t, s \geq 0$) and $\overline{(\beta + W)}$= $\delta_0 \land (\beta + W)$. In this case, we may rewrite (10) as

$$A(t - D) - ((\beta + W) \otimes A) \land (\delta_0 \otimes A)(t)\ \leq\ 0 \qquad (15)$$

and express (12) as

$$D_{work} = h(A, ((\beta + W) \otimes A) \land A)) \qquad (16)$$

where the solution $h(A, A)$ is trivial. Then, using (14) we get:

$$D_{work} = \tilde{h}(a, \delta_0 \wedge (\beta + W))$$

Here a is the minimum flow envelope.

3.5 Maximal delay

Instead, directly reusing the equations (3) and (11), let us determine maximal delay value via explicit computations. Thus way taking into account (6) and, using Theorem 3.1.12 from Le Boudec and Thiran (2019), rewrite (5) as

$$x(t) \geq (A \oslash \beta)(t - D) \qquad (17)$$

Considering the equations (7) and (8) note that they have a solution if and only if the x_{max}, the solution of (9), is greater than the right side of (17). Then

$$(A \oslash \beta)(t - D) - \overline{((\beta + W)}\otimes A)(t) \;\leq\; 0 \qquad (18)$$

Further, using (1) we rewrite (3) as

$$D_{max} = h(x_{min}, x_{\max}) = h(A \oslash \beta, \overline{(\beta + W)}\otimes A\,) \quad (19)$$

Applying the same computational technique as for the case of maximal delay we rewrite (19) as

$$D_{max} = \tilde{h}(A \oslash A, \overline{(\beta + W)}\otimes \beta) \qquad (20)$$

or

$$D_{max} = \tilde{h}(a, \beta_{cl}\,) \qquad (21)$$

Again, If $(\beta + W)$ is sub-additive function and $\tilde{\beta}_W = \overline{(\beta + W)} = \delta_0 \wedge (\beta + W)$, then we can split and rewrite the equation (21) as:

$$D_{max} = \tilde{h}(a, \beta\,) \qquad (22)$$

$$D_{max} = \tilde{h}(a, \beta \otimes (\beta + W)) \qquad (23)$$

The equations (22) and (23) express the maximal value of D for constant W in the sub-additive case. The first equation gives value to the open-loop system when control is not in place.

4. MODEL EXAMPLE

Consider the system represented in Fig. 1 with some detailed parameters.

Let it has strict service curve:

$$\beta(t) = \begin{cases} R(t - T), t > T, R = const \\ \qquad 0, t \leq T \end{cases} \qquad (24)$$

Suppose also that the input flow has a T-SPEC profile:

$$a(t) = \min\,(\gamma(t)_{p,M}, \gamma(t)_{q,L})$$

where $\gamma(t)_{r,b} = \begin{cases} \quad 0, t < 0 \\ rt + b, t \geq 0 \end{cases}$.

The case of $W \geq RT$, then $\beta(t) + W$ is sub-additive, is trivial (see Le Boudec and Thiran, 2019), and we do not consider it here.

The service curve components for open and closing loop cases are shown in Fig.2. Consider the situation of $W < RT$ for two cases: first, the input T-SPEC profile curve $p < R, q < R$ (Fig. 3), and, second, the input curve T-SPEC profile parameters $p > R, q < R$ (Fig. 4). For both cases, we use equations (12) and (19) to compute D_{max} and D_{work} numerically.

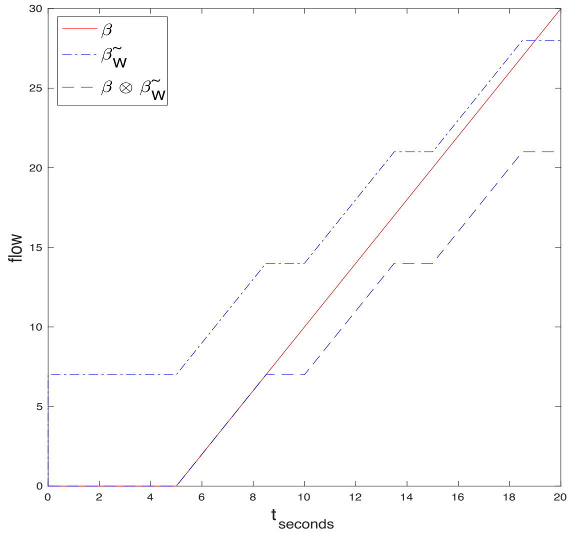

Fig. 2. The closed and open-loop service function of the system $\beta, \overline{(\beta + W)}, \beta_{cl}$, when $W < RT$.

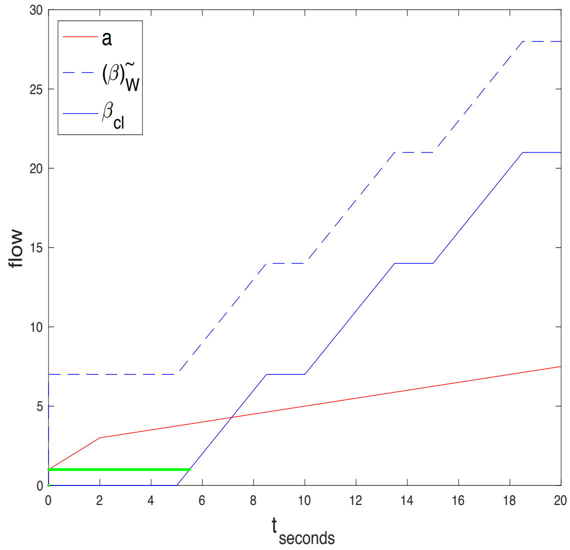

Fig. 3. Closed-loop service function, input (a) and output curve (y) example when $RT < W$ and $p < R, q < R$. The green line shows the maximal horizontal distance between components.

Here (Fig. 3) $D_{work} = 0$ seconds and $D_{max} = 5.5$ seconds.

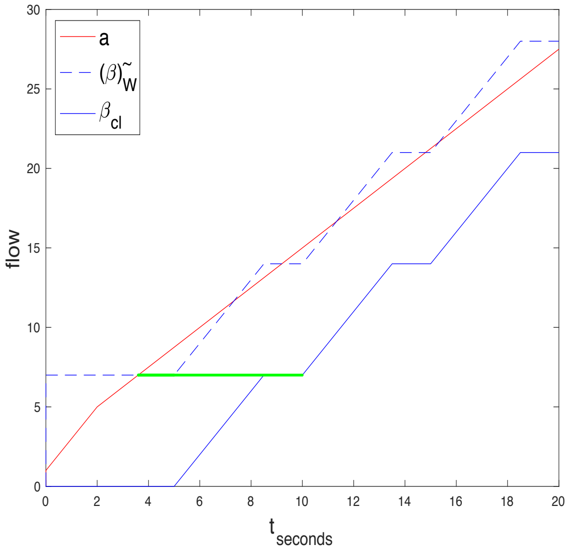

Fig. 4. Closing loop service function, input (a) and output curve (y) example when $RT < W$ and $p = R, q < R$. The green line shows the maximal horizontal distance between components.

Here (Fig. 4) $D_{work} = 1.4$ seconds and $D_{max} = 6.4$ seconds.

It is easy to note that when p increases and $q \to R$, then the maximal and working delay values increase, but they go to infinity when in a case $q > R$.

5. CONCLUSIONS

The work describes delay properties of a system with static window flow control. The motivation of control installation is to counter flows specific to denial of service attacks. The properties of the system are described in the frame of the "Network calculus" approach. We derived the equations describing two types of delay in a system: maximal and working delay. The maximal and minimal properties of the delay and requirement for the buffer system are a common concern for researches, but, unfortunately, the papers dealing with the "Network calculus" approach generally omit the discussion of the working value for delay. However, this value is essential for system designers because it describes normal conditions for the system when static window flow control is acting. Generally, it is easy to see that with input flow growth, the values of D_{work} and D_{max} also grow and so the difference between them gradually vanishes. The system designer shall control the difference of these values to prove the system stability and robustness on a variation of the input flow.

REFERENCES

Agrawal, R, Cruz, R. L., Okino, C., and Rajan, R. (1999). Performance bounds for flow control protocols. *IEEE/ACM Transactions on Networking (7)*, vol. 3, pages 310–323, June.

Baccelli, F., Cohen, G., Olsder, G.J., and Quadrat J.-P. (1992). *Synchronization and Linearity An Algebra for Discrete Event Systems (Wiley Series in Probability and Statistics)*. N.-Y.: John Wiley & Sons, 514 p.

Baybulatov A. A., and Promyslov V. G. (2019). Control System Availability Assessment via Maximum Delay Calculation. *Proceedings of the 2019 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM)*. Sochi: IEEE.

Baybulatov A. A., and Promyslov V.G. (2017). A Technique for Envelope Regression in Network Calculus. *Proceedings of the 11th IEEE International Conference on Application of Information and Communication Technologies (AICT2017, Moscow)*. M.: IEEE, vol. 1, p. 338–341.

Chang C.S. (1998). On deterministic traffic regulation and service guarantee: A systematic approach by filtering. *IEEE Transactions on Information Theory*, vol. 44, pp. 1096–1107, August.

Ciucu F., Fidler M., Liebeherr J., and Schmitt J. (2015). *Report from Dagstuhl Seminar 15112 Network Calculus.* p. 63 – 83.

Cruz R.L. (1991a). A Calculus for Network Delay. Part I: Network Elements in Isolation. *IEEE Trans. on Information Theory*, vol. 37, p. 114–131.

Cruz R.L. (1991b). A Calculus for Network Delay. Part II: Network Analysis Information Theory. *IEEE Trans. on Information Theory*, vol. 37, p. 132–141.

DDoS. (2019). DDoS attack statistics and facts for 2018-2019 *https://www.comparitech.com/blog/information-security/ddos-statistics-facts/*

He. F., and Li, E. (2017). Deterministic bound for avionics switched networks according to networking features using network calculus, *Chinese Journal of Aeronautics*, vol. 30, iss. 6, December, p. 1941-1957

Le Boudec, J-Y., and Thiran, P. (2019). *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet,* Springer Verlag, Online Version of the Book, LNCS 2050.

Masolkin S., and Promyslov V. (2010). The Calculation of Some Properties for Enterprise Network Used in Plants with High Exploitation Risk. *Problemy Upravleniya*, No. 1., p. 47–52. *(in Russian)*

Poletykin, A., Jharko, E., Mengazetdinov, N., and Promyslov, V. (2017). Some Issues of Creating the New Generation of Upper Level Control Systems of NPP APCS, *Proceedings of the 5th International Conference on Control, Instrumentation, and Automation (ICCIA 2017, Shiraz, Iran)*, pp. 78-83.

Syski R. (1997). A personal view of queueing theory, In: *Frontiers in Queueing,* Boca Raton, New York – London – Tokyo: CRC, p. 3–18.